

MACWORLD

Premier Issue \$4.00
Canada \$4.75

The Macintosh Magazine

Macintosh

Apple's Remarkable
New Personal
Computer

An Exclusive
Look Inside the
Macintosh

Word Processing
Tips for
Mac Writers

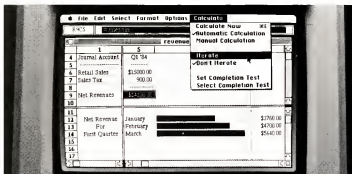
MacPaint's
Amazing
Electronic
Easel

*Steven Jobs,
Chairman of the Board,
Apple Computer*



*Investing with Multiplan
A Tour of the Mac Desktop
Programming Preview
Macintosh Art Gallery
The Making of the Macintosh*

Microsoft's greatest hits are now playing on Macintosh.™



MULTIPLAN. The most powerful spreadsheet on the market.

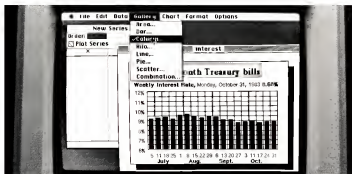
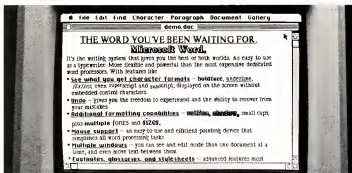
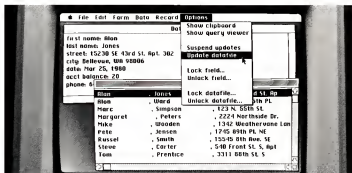


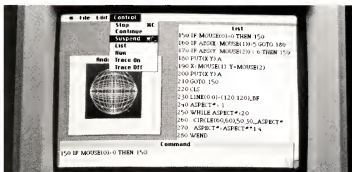
CHART. The first truly sophisticated and flexible graphics program for personal computers.



WORD. Most dedicated word processors can't touch it.



FILE. Finally, a data manager advanced enough to be simple.



MICROSOFT® BASIC. The industry standard. Plus special commands for the mouse and bit-mapped graphics.

Microsoft and Multiplan are registered trademarks of Microsoft Corporation.
IBM is a registered trademark of International Business Machines Corporation.
Macintosh is a trademark of Apple Computer, Inc.

Microsoft was founded on the idea that computers should be simple, easy and intuitive. The new Macintosh environment gives us a great opportunity to prove that idea.

We'll start with four of our high performance programs to make the most of your Mac. And there are more to come. Lots more.

Pioneering like this is nothing new for us. We wrote the first microcomputer BASIC in 1975 and developed the operating system for the IBM® PC.

If you'd like to make the most of your Mac, call 800-426-9400 (in Washington state, call 206-828-8088) for a free brochure and the name of your nearest Microsoft dealer.

MICRO SOFT

TECMAR'S CANDY™ APPLE LINE

BEGINS HERE WITH PERIPHERALS FOR MACINTOSH*



APPLE JACK™ WINCHESTER

Relax with more disk storage.

5 MByte Removable Cartridge Winchester.

APPLE FRITTER™ MODEM

A delicious addition to your Apple.

300 or 300/1200 baud modem with touch-tone decoding, pulse/tone automatic dialer and full voice interface so all the features of a full-feature telephone and modem are provided in one MacIntosh peripheral.

APPLE COBBLER™ PRINT BUFFER

To sweeten the end of a job.

The user no longer needs to wait for a print-out before beginning to program again.

CANDY APPLE™ IEEE488

Interface to the GPIB.

For use in industrial control applications.

*Trademark of Apple Computer, Inc.

Above products will be available in the spring of 1984.

TECMAR

Tecmar Inc. 6225 Cochran Road Solon (Cleveland), Ohio 44139-3377 Phone: (216)349-0600 Telex: 466692

MACWORLD

The Macintosh Magazine



■ State of the Art

16 **A Tour of the Mac Desktop**

Lon Poole

An introduction to the Mac's uniform working environment.

28 **A Mouse in the Hand**

Daniel Farber

Meet the electronic device that communicates your commands to the Mac.

34 **Inside the Mac**

Matthew Douglas

A close-up look at the hardware that gives the Mac its amazing power and speed.

42 **Polishing the Mac**

David Bunnell

Microsoft's Bill Gates discusses the development of the Macintosh.

▲ Review

50 **MacPaint; the Electronic Easel**

Jeffrey S. Young

The Mac's versatile graphics program allows even beginners to draw upon their creativity.

62 **Word Processing's New Look**

Andrew Fluegelman

An extensive review of the capabilities of *MacWrite*, the Mac's native word processing program.

72 **Multiplan Meets the Mac**

Andrew T. Williams

The Macintosh teams up with Microsoft's venerable spreadsheet program.

80 **Apple's Imagewriter Printer**

Lon Poole

What you see is what you get with the Imagewriter.

86 **Macware News**

Edited by Janet McCandless

Announcements of the latest developments in Macintosh hardware and software.

Premier Issue



Hands On

- 94 Investing with Multiplan**
Andrew T. Williams
Constructing a simple stock portfolio valuation worksheet.
- 104 Tips for Mac Writers**
Andrew Fluegelman
Twenty-five word processing hints to help you use *MacWrite* more efficiently.
- 110 SimpleSketch**
Lon Poole
Programming the Mac in BASIC and Pascal.
- 116 The 64K Treasure Chest**
Lon Poole
The Macintosh Toolbox units provide application programmers with a flexible, standard user interface.
- 122 Open Window**
Edited by Andrew Fluegelman
An exchange of Macintosh discoveries.

Photo/illustration credits: Apple Computer, 37, 38, 127; David Bishop, 11, 17, 30, 35, 73, 80, 117; Susan Kare, 29, 57, 136-140; Ed Kasbi, 10, 13, 14, 42, 44; Will Mosgrove, cover; Phil Schermeister, 128-135; Mick Wiggins, 51.



Community

- 126 The Making of the Macintosh**
Edited by Andrew Fluegelman
Reminiscences of the people who created the Mac.
- 136 Macworld Gallery**
Edited by Jeffrey S. Young
An exhibition of Macintosh graphics.



Columns

- 10 A New World**
Andrew Fluegelman
- 13 A Letter from the Publisher**
David Bunnell
- 14 NotePad**
Jeffrey S. Young
Zooming through the Macintosh Window

Index

- 142 Advertisers Index**

MACWORLD

Editorial

Editor-in-Chief: Andrew Fluegelman
Associate Editor: Daniel Farber
Editorial Manager: Janet McCandless
Contributing Editors: Steve Lambert,
Jonathan Littman, Ron Poole, Jeffrey S. Young
Copy Editors: Jeffrey Bartlett, Erfert Nielsen,
Evelyn Spire

Art & Production

Associate Publisher: Jacqueline Potter
Designer: Marjorie Speigelman
Senior Production Manager: Frances Mann
Assistant Production Manager:
Lee R. Trevino
Art Director: Bruce Charonnet
Editorial Production Manager:
Ellyn Hammett
Associate Designers: Dennis McLeod,
Monica Thoros
Design Associates: Darcy Blake, Molly
Winslow McLeod, Donna Share
Telecommunications Manager: Art Wilcox
Advertising Traffic Coordinator: Clare
Bouey

Publishing & Sales

Publisher: David Bunnell
Advertising Sales Manager: Penny Rigby

Circulation

Associate Publisher: Cheryl Woodard
Director Single Copy Sales: George Clark
Subscription Director: Carma Winfrey-Hayes
Dealer Sales Manager: Bill Cox

Communications

Manager: Noreen A. Browning

Special Projects

Director: Susan Keller
Administration Director: Janet McGinnis

Corporate

PC World Communications, Inc.

David Bunnell Chairman

Bartlett R. Rhoades President

Andrew Fluegelman Vice President

Jacqueline Potter Vice President

Cheryl Woodard Vice President

Volume 1, Number 2

May/June, 1984

Macworld™ (ISSN 0741-8647) is published 6 times a year: February, May/June, July/August, September/October, November, and December by PC World Communications, Inc., 555 De Haro St., San Francisco, CA 94107. Effective November 1984 *Macworld* will be published monthly. Subscription rates for the U.S. and Canada are \$30 for 12 issues, \$50 for 24 issues, and \$70 for 36 issues.

Canadian orders must be paid in U.S. funds.

Outside the U.S. and Canada subscriptions must be prepaid in U.S. funds with additional postage at \$12 per 12 issues for surface mail or \$60 per 12 issues for airmail. For subscriber service questions call 800/247-5470 (in Iowa 800/532-1272) or write: Subscriber Services, P.O. Box 20300, Bergenfield, NJ 07621. Application to mail at Second Class pending at San Francisco. POSTMASTER: Send address changes to *Macworld*, P.O. Box 20300, Bergenfield, NJ 07621.

Editorial and business offices: 555 De Haro St., San Francisco, CA 94107, 415/861-3861. *Macworld* is a publication of PC World Communications, Inc. *Macworld* is an independent journal, not affiliated in any way with Apple Computer, Inc. Apple and Mac are trademarks of Apple Computer, Inc. Macintosh is a trademark licensed to Apple Computer, Inc.

The following are trademarks of PC World Communications, Inc.: *NotePad*, *Macware News*, *Open Window*, *Macworld Gallery*, *Macworld View*. *Macworld* is a member of the CW Communications Inc. group, the world's largest publisher of computer-related information. The group publishes 42 computer publications in 18 major countries. Nine million people read one or more of the group's publications each month. Members of the publication group include: Argentina: *Computersworld/Argentina*; Australia: *Australasian Computersworld*; Brazil: *DataNews*; Denmark: *Computersworld/Danmark*; MikroData, France: *Le Monde Informatique*; Germany: *Computerswoche*; Micro Computerswede, PC With, Italy: *Computersworld Italia*; Japan: *Computersworld Japan*; Mexico: *Computersworld/Mexico*; Norway: *Computersworld Norge*; MikroData, People's Republic of China: *China Computersworld*; Saudi Arabia: *Saudi Computersworld*; Spain: *Computersworld/Espana*; MicroSistemas; Sweden: *ComputersSweden*; MikroDatorn, Min Hemadator; United Kingdom: *Computer Management*, *Computer Business Europe*; United States: *Computersworld*, *Desktop Computing*, *Hot CoCo*, *InfoCider*, *InfoWorld*, *Micro Marketworld*, *Microcomputing*, *PC World*, *80-Micro*.

Printed in the United States of America.
Copyright © 1984 PC World Communications, Inc.

SELLING MR. J.R.

WHAT TO EXPECT:

You will find many interactions with Mr. J.R. difficult. You are involved with plans and ideas. Mr. J.R. is concerned with his own inadequacies. You perform well under stress while his performance deteriorates markedly. You think up projects and use initiative to carry them through. He will worry excessively over difficulties and be unable to take action. You welcome the opportunity to be creative and work independently. Mr. J.R. prefers to work under an authority with daily routines involving concrete details. You enjoy status and monetary rewards. He prefers to be recognized for hard work. Exercise caution when dealing with this person. Behave in a reserved manner, use a soft voice and always avoid threats or criticism. Remember, Mr. J.R. is uncomfortable in groups. He is a hard worker, but needs quiet praise.

CUSTOMER PREPARATION STRATEGY

PREPARE INFORMATION IN DETAIL:

You tend to focus on the "big picture" while Mr. J.R. is more concerned with details. Prepare to give a relatively detailed presentation. The actual amount of detail you present depends on the level of detail Mr. J.R. introduces in the needs analysis phase of your meeting.

CUSTOMER CLOSING STRATEGIES

IT'S THE LITTLE THINGS:

Get your customer to agree on the details and end phrases of the sales agreement. It is not necessary to ever mention the sale itself. Focus on delivery, service and color choice as a way to close the deal.

The reason for using the "it's the little things" strategy with Mr. J.R. is that he is very sensitive to establishing a sense of agreement. By agreeing, even on a relatively small aspect of your business, he will feel more secure and more ready to close. Additionally, because Mr. J.R. becomes over-anxious in response to major decisions, working on peripheral details will help him be more relaxed...

Personal power in a printout.

That's what Human Edge™ Software gives you (see full story at right).

This typical excerpt from The Sales Edge™ gives you specific strategies to make every sales call count. The complete printout is a detailed 3-10 page analysis of your business situation.

The Sales Edge is available for the Macintosh® computer.

®Human Edge and The Sales Edge are trademarks of Human Edge Software Corporation.
®Macintosh is a registered trademark of Apple Computer Inc.

Introducing Business Strategy Software: Personal power with a stroke of genius.

It takes more than hard work to stay on the fast track. It takes personal power. Power to communicate. Negotiate. Manage. Power to be a leader.

Personal power isn't just a job skill. It's a career necessity.

Tactics, not numbers.

Introducing Human Edge,[™] the first software family that builds your personal power. Because it helps you manage people, not just words and numbers.

Guided by a vast bank of research on people management, Human Edge asks pertinent questions about you and those your business decisions will involve. It can actually evaluate the human factors that affect your decisions. And give you the best tactics to stack those factors in your favor.

Can software get you a raise?

It even lets you practice with simulated situations, testing different tactics and comparing their outcomes. For the first time,

your IBM[®] or Apple[®] personal computer becomes an expert consultant, preparing you to get exactly what you want. Whether it's asking for a raise, negotiating a contract, or boosting employee performance.

You can't get personal with a cassette.

Human Edge isn't just the floppy disk version of a one-time seminar, management text or cassette.

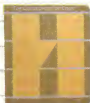
It's a highly creative career tool you tailor to your own management style, your own real-life problems. To guide you through the business jungle with step-by-step strategies. In plain English. In minutes.

Human Edge Business Strategy Software. Because personal power is the way to make your mark.

Call (800) 624-5227, in California (800) 824-7325, for the dealer nearest you. It's the next step on the fast track.

 **HUMAN EDGE**

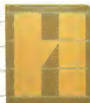
Human Edge Software Corporation, 2445 Faber Place, Palo Alto, CA 94303



The Communication Edge.[™] Find strengths and weaknesses. Improve performance.



The Leadership Edge.[™] Improve powers of group leadership and motivation.



The Management Edge.[™] Evaluate and improve supervisory skills.



The Negotiation Edge.[™] Analyze critical factors. Develop "how to" plan.



The Sales Edge.[™] Sharpen techniques, sensitivity to prospects.

Congratulations Macworld



from the Macintosh™ Team



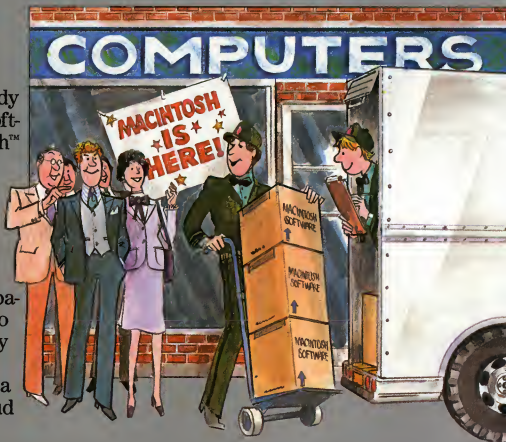
Macintosh is a trademark licensed to Apple Computer, Inc.

How to get your Macintosh programs to market. Quickly.

Call Data Encore. We're already duplicating 3½" microdisk software for the Apple® Macintosh™ system. So we're up to speed and ready to go.

As a subsidiary of Verbatim, the world's leading producer of flexible disks, we're one of the few companies with 3½" duplication capabilities. From raw materials to disks to packaging to delivery to your customers. This total responsibility ensures you of a quality product you'll be proud to call your own.

Why not call us toll-free at 800-872-8778. In California or outside the U.S., call collect (408) 720-7400. Or write us at 585 N. Mary Avenue, Sunnyvale, CA 94086. Because if you want to harvest profits from the new Macintosh faster, you'll want to get your software to market fast.



Data EncoreSM
a Verbatim Company

The Apple Macintosh Book

SHARE THE MAC EXPERIENCE!

Whether you already own an Apple Macintosh or are thinking about buying one, The Apple Macintosh Book provides you with a unique insider's view to this historic advance in computer technology.

Cary Lu, formerly executive editor of High Technology magazine, takes you behind the scenes with this step-by-step guide to how the Mac and its software work. You see exactly what you get—all in the friendly visual style of the Mac itself.

Cary observed the development of the Mac firsthand, sharing the excitement of its evolution with its designers and programmers. In The Apple Macintosh Book he reveals the secrets of the Mac philosophy. And invites you to share the Mac experience.

The Apple Macintosh Book
from Microsoft Press.

Cary
Lu



Coming in March to your local book and computer store.

A New World



Imagine how the fifteenth-century explorers felt as the first news from the newly discovered Western Hemisphere trickled back to Europe. That is the only analogy that approximates the excitement and wonder that I feel about the launching of the Macintosh computer.

As a computer journalist and adventurer, I've had the good fortune to serve as an explorer, historian, and guide during the recent unprecedented expansion of the personal computer world. When I had my first experience with the Macintosh, however, I realized that all my previous explorations had covered only one continent.

No machine has ever been scaled so perfectly for the individual user and for the adventuresome spirit in such an accessible form. I confidently predict that the Mac will change forever our ideas about work and creativity—as well as the way we think about computers. The Mac represents a new frontier in computing, and it's open for all of us to explore, whether we're computer novices or experts.

We've created *Macworld* to guide us in our journeys of discovery. In each issue we'll organize that guide in four sections. In *State of the Art* we'll take a look at the technology that makes the Mac such a remarkable machine and report on new developments that make it even more capable and useful. This issue begins with Lon Poole's "Tour of the Mac Desktop" and Dan Farber's report on the Macintosh mouse. Matthew Douglas' "Inside the Mac" describes the ingenious engineering that enables the Mac to be so compact and powerful, while Microsoft's Bill Gates offers his perspective on the Mac's place in computer history.

Our *Review* section presents in-depth evaluations of Macintosh application programs. In this issue I put *MacWrite*, the Mac's first word processor, through its paces, and Andrew Williams describes how *Multiplan*, a well-known electronic spreadsheet program, fares when translated into the Macintosh idiom. There is almost no precedent for *MacPaint*, the Mac's innovative drawing program. Our review of *MacPaint* is by Jeffrey Young, who brings the perspective of a perceptive computer novice to bear on this unique program. As with all *Macworld* reviews, you'll see that we're not embarrassed

at expressing enthusiasm nor inhibited from making criticisms.

At the end of the *Review* section you'll find a preview of three programming languages and "Macware News," a *Macworld* department that includes descriptions of new hardware and software products for the Mac.

Hands On, the third section of *Macworld*, shows you how to be more productive and creative with your Mac. In this issue we present some tips for streamlining your word processing, a short tutorial in formatting spreadsheets, and some easy exercises in BASIC and Pascal.

A regular part of the *Hands On* section will be "Open Window," which provides a forum for readers to contribute their applications tips and other computing insights. In this issue Lon Poole shows how he created a custom letterhead using *MacPaint* and *MacWrite*. We're looking forward to the "Open Window" expanding as you, our readers, extend the limits of what is possible with the Mac.

Our final section, *Community* includes reports on the ways in which Mac owners

make use of their machines and the ways in which the Mac changes the way we work, study, and express ourselves. For our inaugural issue, we've taken the opportunity to present the story of "The Making of the Mac," as told by some of the people on Apple's Macintosh development team.

A regular part of the *Community* section will be "Macworld Gallery," a continuing exhibition of Macintosh art contributed by readers. Our first exhibitor is Susan Kare, the resident Macintosh artist at Apple who designed many of the graphic images and type fonts that are part of the Macintosh environment. We encourage everyone to take mouse in hand and send us your creations.

Does all this sound exciting? Frankly, I think we've barely established a beachhead in this new world. Beyond the beach are forests to clear, mountains to cross, and wonders awaiting us that we can hardly imagine. In practical terms, I suspect that there are going to be more people using the Mac more creatively to enhance their lives than any computer yet invented. That's what's got me so excited about the Macintosh and the rare opportunity of bringing it to you through the pages of *Macworld*.



Learn to use your Macintosh almost as fast as you can eat one.



Soon after Macintosh joins the Apple family tree we'll have an ATI training program ready to teach you how to use it.

ATI's "How to Use Your Macintosh" training disk and handbook will join our rapidly growing "how-to" series. We'll also have training programs for all the best software available for Macintosh.

All ATI training programs take you through the basics, step-by-step. Our interactive simulation method lets you practice as you learn. It's simple and fun!

Congratulations Macintosh! Together we'll help users- and dealers- bear fruit fast.

Contact ATI, 12638 Beatrice Street,
Los Angeles, CA 90066 (213) 546-5579.
Dealers: Call (213) 546-4725.



ATI Training Power™ is a trademark of American Training International



ATI Training Power™

Learn software the natural way.

Letter from the Publisher



Dear Reader,

Welcome to *Macworld* and to a new dimension in personal computing.

Ideally, computer technology should serve the practical and intellectual needs of people. The Macintosh takes a major step toward that goal. It is an intriguingly elegant machine to which users become attached almost from the moment they switch it on. The Mac reflects the people-oriented perspective of the individuals who created it, and it will have a revolutionary impact on the industry in terms of both hardware and software design.

Not since the introduction of the IBM Personal Computer (and the Apple II before it) has the personal computing scene been so strongly stimulated by a new computer. Before long, there will be thousands of Mac programs and hundreds of peripherals as well as books, training programs, user groups, and even Mac trade shows.

We believe that the only way to provide you with the high-quality publication you deserve as a participant in the Mac revolution is to participate fully ourselves. For the staff of *Macworld*, this is an opportunity to explore the uncharted potential of the Mac, to share our expertise with Mac users, and to join what surely will be a huge Mac support industry.

Macworld, the magazine for Macintosh users, will be your guide through the world of Macintosh computing. We are an autonomous part of the world's largest group of computer publications, including 32 personal computer magazines around the world. As such, we are independent of Apple Computer and have absolute freedom to report both the good and the bad experiences of Mac users. We'll tell you when products don't work as advertised as well as when they do, and we'll keep you up to date on the latest developments in Macintosh technology and applications.

In fulfilling this assignment, we need your help. As much as possible we want *Macworld* to be a reader-interactive magazine. Not only will we be providing you with hands-on tips, product reviews, profiles, and other informative features, we'll also invite you to participate in Mac polls, telecommunication conferences, and other special events. We welcome your letters even when you blast us for something we said or how we said it.

The bottom line is that we are producing this magazine because we believe in it. Thus, if we get lots of letters from you, we'll know we're doing our job, and that can only spur us on to serve Mac users better.

Sincerely,

David Bunnell
Publisher

Letters should be mailed to Letters, Macworld, 555 De Haro St., San Francisco, CA 94107, or electronically to CompuServe 74055, 415 or The Source STE908.

Zooming through the Macintosh Window

First impressions of the Mac by a computer novice



Let me start by presenting my credentials for writing this column: I'm a novice when it comes to computers. Sound odd? Perhaps, but only if you've never encountered Apple's Macintosh computer. They've done their job so well at Apple that within five minutes of sitting in front of the Mac, I was creating documents, pulling down menus, and opening windows on top of windows. And not only that, I was having fun while I did it.

That's right, *fun*. For me, an outsider to the computer generation, the word *fun* seems more suited to the world of arcade-style computer games, not to a computer capable of performing serious applications.

Well, the first time you use the FatBits feature in the *MacPaint* graphics program, see your drawing expanded on screen to the dot-by-dot level, and watch a reduced image of the entire drawing in the corner of the window change as you replace each dot, I guarantee that you'll be having fun. Or maybe you'd prefer to select one of the many type fonts available in *MacWrite* to write your next memo—I can't wait to send someone a memo in Old English. And perhaps you'd like to place a drawing of your dog or cat in the middle of a

note. Whatever your preference, the horizons of what is possible with a computer have suddenly expanded.

Most computers I've seen that are capable of useful applications are "solemn machines," with a businesslike look and no-nonsense approach to life. You have to be extremely motivated (usually by a job, or perhaps by a profound fascination with gadgetry and technology) to memorize the multiple commands and procedures necessary to make the machine work. And maybe once in a while you can run a clunky imitation of a video-arcade game, pretending that you are having fun while knowing that if this entertainment were from the not-too-distant past, it would have come packaged in a nice cardboard box with the ominous phrase "fun for all ages" emblazoned upon it.

You don't need an encyclopedic manual to get started with the Mac. Basic word processing operations, such as changing the line spacing of a document, are so simple and efficient that even the most computerphobic people will feel less intimidated by the machine.

With other computers I've used, changing the line spacing of a document involves typing a series of cryptic commands. Forget those nightmares when you use the Mac. With the Macintosh mouse in hand, you just

drag the pointer across the screen to the small icons above the document that graphically depict either single-, double-, or line-and-a-half spaced text, and click on the icon with the spacing you want—the document is instantly formatted according to your selection.

Don't like that format after you're through with the letter? Want to go from double to single spacing? No big deal—you select the text by clicking the mouse button at the start of the letter and dragging the pointer to the end of the letter. Next, drag the pointer to the icon showing the format you now want and click the mouse button. The entire body of text is rearranged before you can blink. Perhaps you don't like the new arrangement once you see it. No problem, just click the mouse button on the icon representing the original format or choose the Undo command from the Edit menu, and the document instantly reappears in its original format.

Sound too easy? Maybe, but I for one always figured that some of those brilliant computer people would eventually come up with a way for all of us noncomputer types to use these machines without drastically changing our lives—that is, without learning some new language of keyboard hieroglyphics.

The Macintosh concept of the desktop environment, with files and folders, windows on top of other windows, and commands that can be activated with a simple action of your wrist and the mouse, is a powerful analogy to the way those of us still living in the precomputer age organize information. I have files in drawers and others scattered across the top of my desk (more on top than I care to admit). I have a large collection of pens, pencils, scissors, glues, paper clips, thumbtacks, cellophane tape, erasers, rubber bands, and diverse notepads cluttering my desk. I open or close files as they are needed, take information from one file and put it into another, create new files, and throw files in the trash.

The overall organization of the Mac desktop is similar to my work habits, and it replaces many of those archaic desk accessories with more efficient electronic devices. I feel more organized and comfortable with my "electronic" desktop, especially the way in which I can control the entire set of operations on the screen with the mouse and one hand—I lean back, stretch out my arm, and edit with ease. No more hunkering over the keyboard and typing draft after draft of not-quite-perfect prose.

And perhaps more importantly, I feel in command of the Mac. I don't think I can make that point strongly enough: from the moment I switched on the computer (without the aid of any instructions) and encountered its desktop-like screen, I knew that I was in the presence, not of an enemy machine to be saddle broken and conquered, but of an intellectual friend that was going to take care of certain heretofore monotonous and unwieldy tasks.

What I didn't know (and certainly didn't expect from a computer) was that the machine would also stimulate my creative juices in a way I hadn't encountered since I first discovered the opposite sex in the eighth grade. For me, using the Mac is like wandering into an overstocked playpen of the mind, where each path leads to another path and each discovery produces more avenues to explore.

By expanding the skills I have at hand and the tools I use, the Mac has expanded my artistic and intellectual boundaries. For instance, I haven't drawn a picture for years, primarily because I'm so hopeless an artist. Using *MacPaint*, however, I'm rediscovering the language of graphic images—I feel like an electronic Picasso.

What will be the final result of that new perspective? I don't know, and the ingrained habit

of thinking in mostly verbal terms will take time to reverse. But I'll venture a guess that within a year I'll be thinking "visually," taking into account the possibilities for graphics in all my work. That expansion of my "mental palette" is an exciting prospect. Where it will lead is the kind of open-ended question that the Mac asks so well—and so far refuses to answer with any finite limit.

The Mac may become the first truly mass-market computer.

The Mac may become the first truly mass-market computer simply because it doesn't take any special language to make it work. Whoever you are, whatever you do, you don't need to be computer literate to use it. And yet the simplicity of the Mac hides a more advanced computer than has ever before made it to the personal computer consumer marketplace.

What do the innovations of the Mac portend for the future? What are kids going to be like when they never have to learn how to use a pencil or an eraser? Will the Macintosh or its successors make any lasting

difference, or is the Mac just another mutation of the glorified electronic beast of the late twentieth century? These are questions that will be answered in the years to come.

However, my intuition tells me that the Mac will change the way people think about computers. There are millions of people like me who don't want to have to go very far out of their way to use a computer. The Mac will work for us because at its root, it's more powerful than most of those other personal computers (simple elegance, such as the ease with which the Mac works, is always the most difficult thing to produce), and it reflects the practical needs of people, not technically oriented designers.

Let's see, if I select the paintbrush from the tool palette, choose that herringbone pattern I love so much, and start to fill in the series of concentric circles I just created... well, I don't know if it will give Picasso a run for his money, but it is certainly all mine.

Drag the pointer, open a new window, and take a look out into the future. Maybe even climb through that window. The Macintosh computer has arrived.

A Tour of the Mac Desktop

Lon Poole

With the Macintosh, Apple has added a new dimension to computing. Based on the concept of a desktop working environment, the Mac allows you to do more with a personal computer—and more importantly, do it more easily and naturally—than ever before.

Imagine driving a car that has no steering wheel, accelerator, brake pedal, turn signal lever, or gear selector. In place of all the familiar manual controls, you have only a typewriter keyboard.

Any time you want to turn a corner, change lanes, slow down, speed up, honk your horn, or back up, you have to type a command sequence on the keyboard. Unfortunately, the car can't understand English sentences. Instead, you must hold down a special key with one finger and type in some letters and numbers, such as "S20:TLA35," which means, "Slow to 20, turn left, and accelerate to 35."

If you make typing mistakes, one of three things will happen. If you type an unknown command, the car radio will bleat and you will have to type the command again. If what you type happens to be wrong but is nevertheless a valid command, the car will blindly obey. (Imagine typing A95 instead of A35.) If you type something the manufacturer didn't anticipate, the car will screech to a halt and shut itself off.

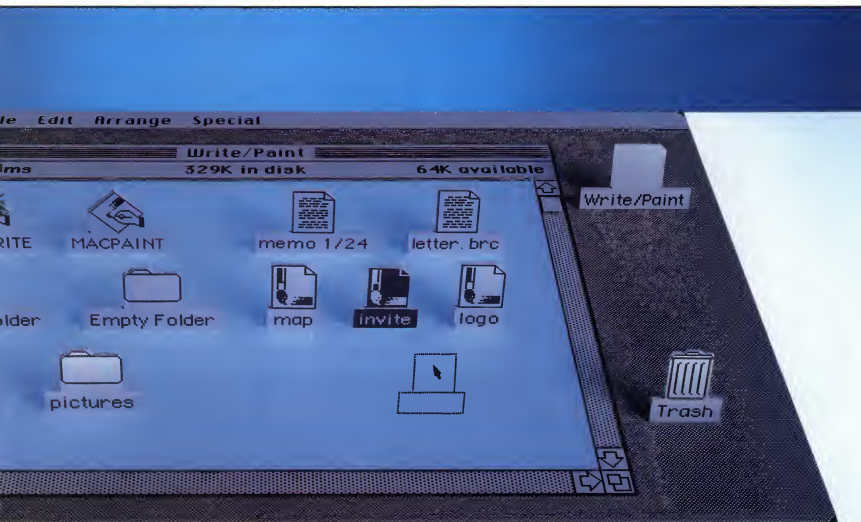
No doubt you could learn to drive such a car if you had sufficient motivation and determination. But why bother, when so many cars use familiar controls? Most people wouldn't.

Most people don't bother to use a personal computer for the same reasons they wouldn't bother with a keyboard-controlled car. Working on a computer isn't a natural skill, and the benefits hardly seem worth the hassle of learning how to get work done in an unfamiliar environment. If you make a typing mistake, the computer may do nothing, tell you it doesn't understand, do the wrong thing, shut itself down, or destroy all the work you've done and then shut itself down. Who cares if the machine is theoretically thousands of times more efficient than pencil and paper? If using the machine rattles you so much that you can't get anything done, it is in fact less efficient and may waste more time than it saves.



.....

The Mac desktop contains small pictures, or icons, that represent the programs and documents stored on the disk inserted in the Mac's disk drive.



What if a computer could let you work in a familiar environment, similar to the way you work at your desk? You could put things you wanted to work with on top of the "desk," move them around, put documents into folders or files, and even throw things into the trash. This description accurately fits the working environment of Apple's Macintosh computer. The things you work with on your desk appear not as words and numbers in regimented lines, but as graphic objects located on the Mac screen.

Icons

The Mac desktop, being somewhat smaller than the average desk it models, doesn't have room enough for life-sized objects. At first, objects appear on the Mac desktop as small pictures called *icons*. On the Mac, an icon is a symbol for some concept or object. For example, when you switch on the Mac and insert a disk, the screen shows two disk-shaped icons and a trash can (see Figure 1). As a graphic image, an icon can remind you about what it represents better than words alone.

Each icon represents a specific collection of information. To avoid ambiguity, icons also have labels. The disk icon (labeled Write/Paint in Figure 1) represents the documents and programs stored on the disk inserted into the Mac's disk drive. The dimmed disk icon labeled Alternate Disk is used for copying files from one disk to another, and the trash can icon labeled Trash holds documents and programs waiting to be purged from the disk.

Pointers and the Mouse

On a real desktop, you move things around. You may work with one document or file for a while, switch to another, do some minor calculations, check the time, and then create a new file. When you finish working on something, you want to put it away somewhere convenient (such as in a file drawer) so you can retrieve it later. The Mac lets you do all these things, but the things you work with exist as graphic images on the Mac's electronic desktop.

Since you can't touch things on the Mac desktop, you need some form of remote control. The mouse is the key to working on the Mac desktop. Sliding the mouse on a smooth surface moves a pointer on the screen. Slide the mouse in any direction—up, down, sideways, or diagonally—and the on-screen pointer will move the same distance in the same direction (see "A Mouse in the Hand" for an in-depth view of the Macintosh mouse).

Moving the mouse moves the pointer, but pressing the mouse button makes things happen. For example, you can move an icon by placing the pointer over it, pressing and holding down the mouse button, and then dragging the icon to a new location. The moment

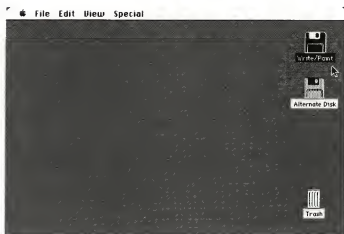


Figure 1

The Mac desktop. The Write/Paint disk icon represents the documents and programs stored on the current disk; the Alternate Disk icon is used for copying files from one disk to another; and the trash can icon holds discarded documents.

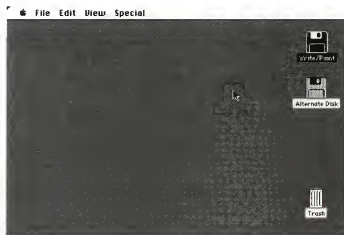


Figure 2

Dragging an icon. As you hold down the mouse button and move the pointer, an outline of the selected icon moves across the screen.

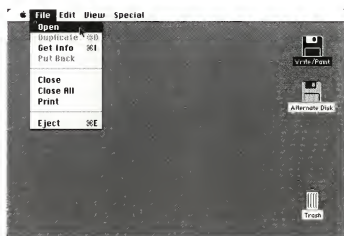


Figure 3

The File menu. To select a command, drag the pointer down the menu and release the mouse button when the command is highlighted.

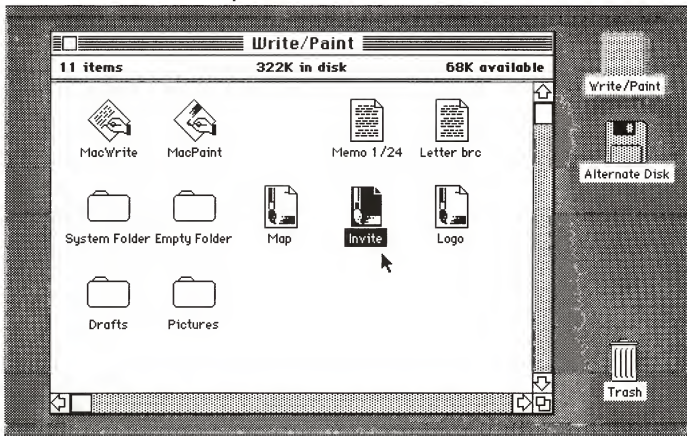


Figure 4

The disk window. The icons represent the documents and programs on the current disk.

you press the mouse button, the icon is highlighted. As the pointer moves, it drags an outline of the icon and its label along. The outline shows you where the icon will appear when you release the mouse button (see Figure 2).

Moving the pointer over an icon and pressing and releasing the mouse button highlights the icon. This mouse action, called *clicking*, selects the object but does nothing except highlight it; you still must specify an action. In other words, you must give a command.

Menus

On most computer systems, you issue a command by typing arcane words or symbols on the keyboard. Remembering such commands is difficult enough from day to day. Go to Hawaii for a week, and you can plan on a session with the manual when you return.

The Mac never forces you to remember command words or type commands on the keyboard. All Mac commands are listed in menus, and you choose them with the mouse. Don't let other menus you have seen or heard about prejudice you against Mac menus. Most people say menus are great when you're learning something, but they slow you down too much when

you know the ropes. Not on the Mac. Most people complain that menus take over the screen, making the information they're acting on invisible. Not on the Mac. Most people say even with a menu you still end up typing in a code number or letter. Not on the Mac. Mac menus are unobtrusive and fast, and require no typing.

The Mac desktop has five primary menus—Apple, File, Edit, View, and Special—that stretch across the top of the screen. These menus provide all the commands for organizing and working on the desktop. You can do everything from opening and closing files to rearranging icons. The Mac hides its menu commands under the menu titles. When you move the pointer over one of the menu titles and press the mouse button, a list of commands drops down from the menu bar, temporarily overlaying a small part of the screen.

The Apple menu (represented by the apple symbol) contains a selection of desktop accessories and controls (see "Desktop Accessories" for a comprehensive look at the Apple menu options). The File menu lists commands for working with files (see Figure 3),

Desktop Accessories

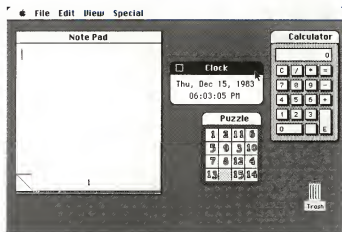
Daniel Farber

You've probably noticed the apple in the upper-left corner of the desktop menu bar and wondered what it does. If you press and hold down the mouse button on this icon, a list of options will appear instantly. These are the soon-to-be-well-known Macintosh desk accessories, which are manipulated with the mouse and are available as part of any application program for the Mac. The desk accessories provide a number of useful functions similar to those performed by objects on your "real" desktop. The Mac also allows you to create your own desk accessories. The following are brief descriptions of each accessory.

● **Calculator.** The hand-held calculator rose to prominence in the 1970s, joining pencil and paper as necessary tools of every student, consumer, and businessperson. The Mac partially replaces those "concrete" objects, not only by reducing the need for pencil and paper, but also by providing a basic four-function, hand-held calculator. You can use the mouse to click on the numbers and function symbols, or you can perform calculations using the keyboard or the numeric keypad. Results of calculations can be cut and pasted into another desk accessory or into any document, and numbers from a document can be pasted into the calculator's display.

● **Clock.** We live in a time-conscious era; we have clocks on our desks, walls, stoves, cars, and wrists. The Mac also has a clock that can be placed on the desktop. It gives you the current date and time and has an alarm that can be set with the mouse. You can copy the date and time and paste them into other documents or accessories.

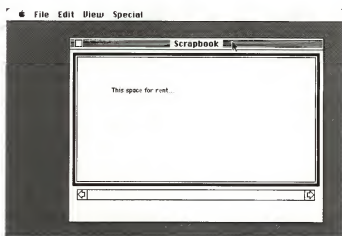
● **Key Caps.** The Macintosh comes with a set of optional characters. The Key Caps accessory shows the characters you can type while holding down either the Option, Shift, or Caps Lock key. To get the character you want, you can click it while you are in the Key Caps accessory or type the particular key combination on the keyboard. Characters typed in the Key Caps window can be edited, and you can cut and paste them into any document or desk accessory.



The NotePad, Clock, Puzzle, and Calculator have functions similar to those "real" objects on your desktop.



Key Caps displays the Mac's optional character set.



The Scrapbook stores images or text that you use frequently.

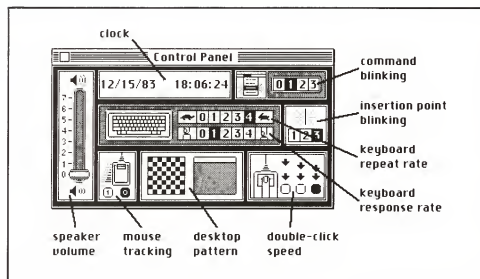
● **Puzzle.** When you need some relief from a difficult computing task, you can select the Puzzle option. The Apple menu puzzle is an electronic simulation of the familiar plastic tile number scramble. You click on the "tiles" to move the numbers into sequential positions. Whenever you close the Puzzle the tiles are rescrambled.

● **NotePad.** Taking notes or memos has become an essential action in our fast-paced society. People carry around scraps of paper covered with phone numbers and financial figures, and appointment books stuffed with daily reminders. You may not be able to carry your Mac around, but you can keep your notes and messages in the NotePad. It holds up to eight numbered pages of text that are automatically saved on the disk in a special NotePad file. NotePad text is editable, and you can cut and paste text into and out of it. Clicking on the turned-back corner of a page brings up the next page, and clicking on the bottom-left corner brings up the previous page. This accessory is handy for jotting down notes that you want to keep separate from a document you're working on. If you are in the midst of a spreadsheet application and suddenly get an idea for the final chapter of your mystery novel, jot it down in the NotePad.

● **Scrapbook.** People often use a graphic image or a portion of text for several purposes. The Mac has a Scrapbook file for collecting images and text you want to keep available. For example, if you have a standard letterhead or logo you use frequently, you can copy or cut it into the Scrapbook. When you want to paste that element into a document, scroll through the contents of the Scrapbook file until the image you want appears and then click the Paste button in the window.

● **Control Panel.** If any feature typifies this machine's design as a user-scaled computer, it's the set of custom options available in the Control Panel. Calling up this subprogram permits you to control nine system functions at any time. Two of them are fairly straightforward; one lets you adjust the volume of the internal speaker, and another resets the time and date on the Control Panel clock.

Some of the other controls are truly innovative. You can specify both the rate at which the keys repeat (clicking the lower



The Control Panel allows you to control nine system functions at any time.

numbers causes a slower repeat rate) and the keyboard response rate (the length of time a key must be held down before it produces a repeating character). Clicking a low number means you'll have to hold down the key longer to get a repeating character; clicking 0 disables the character repeat feature.

You can also control the blinking rate of the insertion point (vertical bar) and the command selection bar. When you drag the pointer over a command in a pull-down menu, for example, the command is highlighted. When you release the mouse button, the highlighted command blinks according to the level set in the Control Panel. (Setting the blinking rate at 0 disables the feature.) These two options help users to control the visual cues on the desktop.

Two controls govern the actions of the mouse. The mouse tracking control determines whether the movement of the mouse on the desktop will produce a corresponding one-for-one, dot-by-dot movement of the pointer on the screen. Setting the mouse tracking control at 0 maintains a constant, dot-by-dot pointer speed. Setting the control at 1 causes the pointer to move farther, skipping every other dot on the screen, but only when you

speed up the mouse movement. Having the faster mouse tracking speed in operation facilitates taking action on the screen.

The other mouse control determines what time span qualifies as a double-click. This may sound like an abstract concept, but being able to control double-clicking speed lets users adjust the computer to their styles, rather than vice versa. The setting with the arrows closest together is the most rapid double-clicking speed.

The desktop pattern control feature has a purely aesthetic function. It allows you to specify the pattern used for the desktop. The default setting is a neutral, 50-percent gray pattern. You can scroll through more than 30 patterns, ranging from solid black or white to crosshatchings, brick patterns, and wave forms. You can even customize your own pattern by editing the pattern in the FatBits portion of the pattern window.

The Macintosh might well have been successful without the Control Panel. The fact that Apple programmer Andy Hertzfeld took the time to design it—and that Apple put his creation into the final product—reflects the overall concept of the Macintosh style of computing.

■ ■ ■ ■ ■ Daniel Farber is
the Assistant Editor of Macworld.

and the Edit menu contains basic editing commands. The View menu lets you organize your files by icon, name, date, size, or kind on the desktop. The Special menu includes two options: Clean Up arranges the icons in orderly rows and columns, and Empty Trash deletes files permanently from the disk.

Menus operate as if they were spring-loaded. As long as you hold down the mouse button, the menu choices (commands) stay in view; when you release the mouse button, the menu choices disappear back under the menu bar. While you hold down the mouse button, you can drag the pointer down the menu. Each menu choice is highlighted temporarily as the pointer passes over it. You choose a command from the menu by releasing the mouse button when the command you want is highlighted. Also, some of the menu commands have keyboard equivalents, which are listed next to the corresponding command.

All of the available commands appear in black type. Sometimes it doesn't make sense to use some of the commands. Commands that are out of context in any particular situation appear in gray, or dimmed, type. They are not highlighted when you move the pointer over them, and you cannot choose them. In the File menu, for example, you cannot Close or Print a document unless you first Open it.

Windows

When you want to look at the information that one of the icons represents, you open a window. To open the disk icon, for example, you first select the icon by clicking the mouse button while the pointer is over the disk icon. The icon is highlighted to confirm that you have selected it. Next, you choose the Open command from the File menu. An outline zooms out of the icon and the screen almost fills up with a rectangular "window" containing icons that represent the documents and programs on the disk (see Figure 4). The selected icon becomes hollow (all white) to show that you have opened it, and the disk icon's name appears in a *title bar* at the top of the window. The line below the title bar gives information including the number of files, the amount of disk space they take up, and the amount of disk space available.

A more efficient way to open an icon is to *double-click* the mouse button (quickly press and release it twice); this action selects the icon and opens a window.

Some of the icons represent folders that can contain other programs and documents, similar to file folders on your office desk that combine separate files. You can see the contents of a folder by selecting and opening that folder. A new window will appear on the desktop, displaying the icons that represent the files stored in the folder (see Figure 5). You can store folders within folders and use them to organize your files so that windows don't get cluttered with too many documents.

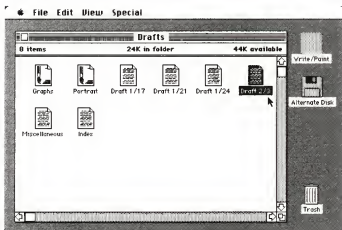


Figure 5
A window showing the contents of a folder. Each folder contains documents and programs, which in turn can contain other documents and folders.

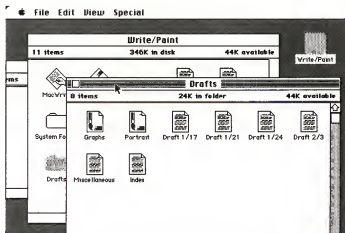


Figure 6
Overlapping windows. You can open several windows at once; the topmost window is the active window.

The Mac lets you open several windows simultaneously. Select another icon, choose the Open command from the File menu or double-click on the selected icon, and another window zooms into existence. Each new window you open overlaps the existing windows. You may see the edges of existing windows sticking out underneath the new window, or the new window may completely hide everything under it. Windows can also cover up the icons on the Mac desktop (see Figure 6).

The window on top, or foremost window, is called the *active* window. You can bring any window to the top and make it the active window by putting the pointer anywhere on it (even an edge that's sticking out behind another window) and clicking the mouse button. You can remove the active window from the Mac desktop by choosing the Close command from the File menu. The icon that the window came from sucks the information back, the window disappears, and the icon resumes its normal appearance.

You can also move windows around on the Mac desktop. If you place the pointer over the title bar of a window, press and hold down the mouse button, and slide the mouse, a flickering outline of the window is dragged on the desktop. Let go of the mouse button, and the window jumps to the new location. When you move a window by this method, it becomes the top-most window. However, holding down the \mathbb{H} key while you drag a window allows you to move the window without disturbing its relative position in the pile. This feature is an example of an "advanced" desktop management skill that you soon learn after a few work sessions with the Mac.

Sometimes windows get buried. Unfortunately, there's no way to get a side view of the Mac desktop to see what might be under the foremost window. But you can always relocate windows or change their sizes to uncover the ones underneath.

Window Controls

Think about the appliances you use. They have pushbuttons, knobs, dials, and other types of controls. Because the Mac desktop is a general purpose information processing appliance, it needs many different controls. You've seen the way elaborate stereo systems bristle with knobs, buttons, and dials. Imagine adding a television, telephone, and pocket calculator to that collection. Pretty intimidating, but nothing compared to what the Mac would look like if it had separate controls for everything it did.

Most computers handle the control problem by overworking the keyboard. A few add some so-called function keys, but you have to be a double-jointed

NASA rocket control specialist to use them effectively. The Mac displays controls to suit the situation. You use the mouse to activate buttons and adjust control knobs displayed on the screen. For example, the disk window, like most windows, has several controls built in (see Figure 7). The Mac displays the controls only when the window is active, however. To close an active window, you can click the mouse button while the pointer is over the *close box* at the left side of the title bar. Clicking in this displayed box has the same effect as choosing the Close command from the File menu. (Savvy Mac users will quickly discover many time-saving shortcuts.)

The small box displayed in the lower-right corner of most active windows gives you control over the size of the window. To make the window narrower, use the mouse to drag this *size box* to the left. To make the window wider, drag the size box to the right. Drag the size box up and the window gets shorter; drag it down and the window gets taller. Drag the size box on a diagonal to change both height and width simultaneously.

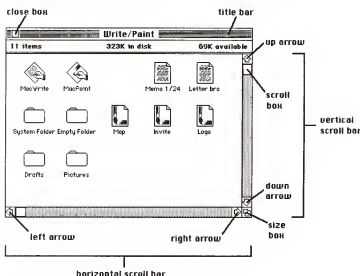


Figure 7
Window controls enable you to change a window's size, scroll vertically or horizontally within it, or close an active window using the mouse.

ly. Changing the window size does not change the size of what's displayed, it just changes the amount of information you can see at once (see Figure 8).

Sometimes a window doesn't show all of its contents, even if you've fully extended its size. Fortunately, the *scroll bar* controls let you scan back and forth over the available information. Most windows have two scroll bars. One, located on the left edge of the window, controls up-and-down movement. The other, located at the bottom of the window, controls side-to-side movement.

Of the many ways to use scroll bars, the simplest is to click the arrow that points in the direction you want the window to move over the information. (Actually, the window stays put on the screen and the information moves under it, but the effect is the same as if the window had moved in the direction of the arrow used.) If you press and hold the mouse button instead of just clicking it, the window keeps moving. As the window moves, a small white box, the *scroll box*, also moves. The scroll box gauges the window's position relative to the top and bottom, left and right edges of the screen.

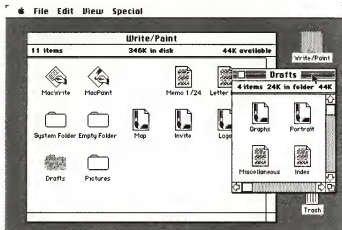


Figure 8
Small and large windows. Changing a window's size affects the amount of information visible at one time.

Dialog and Alert Boxes

The Mac takes the unexpected in stride. When something unusual happens, it displays a special window to inform you of the exceptional circumstances. You may have to click some buttons to cancel or continue an action, manipulate some other controls, or even type a name on the keyboard. These special windows, called *dialog boxes*, appear only for the purpose of getting supplemental information from you, information needed to proceed with the task at hand (see Figure 9). If the special window appears because of some potentially dangerous situation (such as when your disk is almost full), it is called an *alert box* (see Figure 10). The appearance of an alert box may be accompanied by one or two beeps from the Mac's speaker.

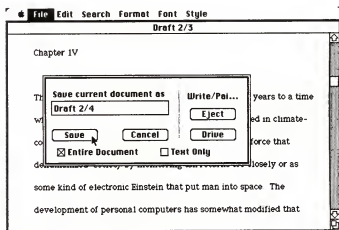


Figure 9
A dialog box appears when the Mac needs additional information to proceed.

Application Programs

The commands available from the menu bar and the controls displayed on the screen vary depending on the program you use. Every program has an icon, and opening that icon starts the program. Starting a program opens a window in which you can display a document of your choice. You can create a new document or call up an existing document from the disk. The program also displays its own menu bar and controls that you can use to inspect and change the document.

Generally speaking, you can also start a program by opening a document that you created with it. You can start the *MacWrite* word processing program, for example, by opening a letter you wrote using it. You do this by double-clicking on the icon representing that document.

On the Mac, however, you can run only one program at a time; you can't open a *MacWrite* window at the same time you have a *MacPaint* window open. Apple had to leave something for the Lisa to do better. (You can, however, open and use a desk accessory program while you are using another program, and you can place *MacPaint* drawings into *MacWrite* documents as explained below.)

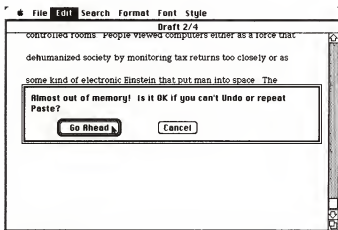


Figure 10
An alert box warns of a potentially dangerous situation.

The Universal Interface

Although the specific commands and controls are different from one program to the next, all Mac programs adhere to certain conventions, including the use of icons, windows, menus, and the mouse. Once you discover a way to do something in one program, you can apply the same principles in other Mac programs (see "The Mac Way" for more information about the Macintosh user interface). For example, the method for moving information from one part of a document to another is uniform, no matter what kind of information is involved. That procedure, called *cut and paste*, is done entirely with the mouse and the Edit menu (see Figure 11). The steps are as follows:

1. Select the information to be moved.
2. Choose the Cut command from the Edit menu.
3. Select the insertion point.
4. Choose the Paste command from the Edit menu.

menu.

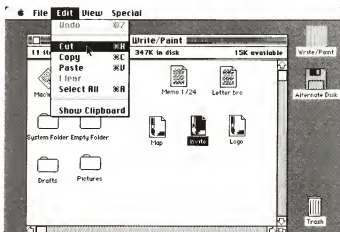


Figure 11
The Edit menu. The Cut and Paste commands are used to move text or graphics from one part of a document to another.

You can copy information by choosing the Copy command instead of the Cut command in step 2. Copy and paste is completely analogous to cut and paste but does not delete the original information. Either way, the procedure works if the information comes from one document and goes to another, even if the documents were created by different programs. It doesn't matter whether you transfer text to text, text to a drawing, a drawing to a drawing, or a drawing to text.

A Personable Computer

The popular notion of computers dates back 15 or 20 years to a time when computers studded with flashing lights hunkered in climate-controlled rooms. People viewed computers either as a force that dehumanized society by monitoring tax returns too closely or as some kind of electronic Einstein that put man into space. The development of personal computers has somewhat modified that perception. Progressive thinkers now regard computers as impersonal machines, just so many keys to press and commands to remember. Rudeness is probably the only characteristic anyone would anthropomorphically attribute to them.

The Mac, on the other hand, is a responsive, active, engaging information processing appliance that is incidentally a computer. It doesn't intrude. It is quiet, takes little space, and doesn't ask you to remember anything. When you insert a disk, the screen fills with icons representing objects you find on or around your desk. These graphic images soon become very familiar to you, like the actual papers, folders, trash can, and documents they imitate. You copy a document, choose a command, drag an icon, cut and paste a paragraph, sketch an illustration, sum a column of numbers, all by manipulating objects on the screen with the mouse. You tell the Mac what to do and it reacts, not the other way around. You think, "This electronic desktop may not be so absurd or useless after all." Finally, a computer that doesn't act like one.



Macintosh
Apple Computer, Inc.
20525 Mariani Ave.
Cupertino, CA 95014
800/538-9696; in California
800/662-9238

List price: \$2495 (includes Macintosh, keyboard, mouse, owner's manual, system disk, blank disk, power cord, programmer's switch, two Apple decals, and tutorial disk and audio cassette)

The Mac Way

Most computers don't have a uniform working environment, or universal interface, like the Macintosh. Consequently, every application ends up with a different way of doing the same thing. Trying to remember those various ways can be very frustrating. All of the applications currently available on the Mac share the same working environment. If you learn how to edit text using *MacWrite*, for example, you can expect the same basic techniques to work in *MacPaint* and *Multiplan*. Of course, *MacWrite* will allow you to do more with text than will *MacPaint* or *Multiplan*, but all Mac applications have a uniform approach to fundamental tasks.

Exploring a new application will not be an intimidating or frustrating experience because you will be able to apply what you already know from working with other applications. You may start by learning how to draw using *MacPaint* and find it so entertaining that you will actually produce something useful, such as a map to your house or a letterhead for your personal stationery. That success will encourage you to try writing letters and memos using *MacWrite*. Because that experience was so painless, you'll brave *Multiplan* to create a budget or an expense account. The working atmosphere is so familiar and consistent from one application to the next that you will find yourself doing things you never thought you would even try, with or without a computer.

All this talk may seem a bit odd or even pointless if you already use another personal computer and feel comfortable with it. You probably have some handy reference cards listing all the commands for the programs you use, so if something should slip your mind, you can easily look it up.

But admit it, aren't there times when you growl in anguish as your computer deletes the wrong file because of a typo-

graphical error? (Such errors are impossible on the Mac because you never need to type commands.) Don't you wish you could move the cursor diagonally, not just up, down, and sideways? (With the mouse you can move quickly from one point directly to another.) Wouldn't you like to get rid of the pencil and paper you use to remind yourself of which seven cells you want to add together for a total in a spread-

All of the applications currently available on the Mac share the same working environment.

sheet? (All you do on the Mac is point at each cell or at a range of cells, and the program remembers for you.) And suppose you had a word processor you could use without having to remember dozens of control codes. Who knows, you might be tempted to type your own letters and memos, rather than writing them out long-hand and having someone else type and retype them. (You never have to use a control code in *MacWrite*.)

Switching to the Mac from another computer will be far less trouble than trying to remember the multitude of commands for the applications you have already learned. You have a head start over computer newcomers because you know word processing concepts such as word wrap, page headers, boldface characters, and justified margins, or spreadsheet concepts such as column replication, summing a range of cells, and dollar formats. All you have to do is learn how to get around on the electronic desktop. (Don't forget that what you learn about in one application will stand you in good stead in another.)

It's not surprising that the first applications for the Mac use consistent methods and can share information. What about the applications to come? Apple is encouraging independent program developers to create new applications for the Mac. It will be up to the programmers who develop new applications to observe the rules and provide Mac users with a consistent, familiar working environment. Fortunately, they have at least three good reasons to do so:

- Mac owners will be more likely to buy software that they find familiar and consistent and hence easy to learn. Given the choice between two programs with comparable features, people will always pick the one that is similar to the programs they already have. *MacPaint*, *MacWrite*, and *Multiplan* will set the de facto standard working environment against which all other programs will be measured.

- Program developers will not have to invent, describe, and justify their own unique operating environment. Apple has already put years of research, development, and testing into the design of the Mac desktop model. It is general and flexible enough to work in almost any application.

- The tools that programmers need to create the prescribed settings and trapings of the desktop environment are built into the Mac. It costs programmers nothing to use them. In fact, using them means less programming, not more, and a more reliable program that uses less memory can be finished and bring in revenue sooner.

If program developers use their imaginations to come up with innovative applications while staying consistently within the Mac working environment, Apple's dream of a universal interface will become a reality, and the computing power of the Mac will be accessible to a wider range of users.

A Mouse in the Hand

Daniel Farber

The Macintosh desktop environment was designed to make working on a personal computer easier and more productive. And the key to working in that environment is the mouse. This little, hand-held device puts you in control of all the Mac's unique features.

The juxtaposition of a mouse and a computer on a modern office desktop or your cozy office at home might seem strange indeed. Is it some kind of marriage between high technology and the rodent population, or an example of the arcane Silicon Valley sense of humor?

Actually, the mouse is not new to computers. It has been around since the early 1960s when Douglas Engelbart, an associate at the Stanford Research Institute (SRI), created a small, mouse-shaped device with three protruding control buttons that looked like ears and a cable connecting the device to the main unit of the computer that looked like a mouse tail. Engelbart's objective was to design interactive computer aids that made using the machine more natural and less alienating.

Basically, the mouse works as a combination cursor-movement and all-purpose function key that allows you to operate a computer independent of the keyboard. With the Macintosh mouse, for example, you slide the device over a smooth surface with your hand, and the graphic pointer on the screen precisely follows its movement. A single button on top of the mouse allows you to select information or objects and activate commands. Essentially, the keyboard is a supplement to the mouse in the Macintosh desktop environment.

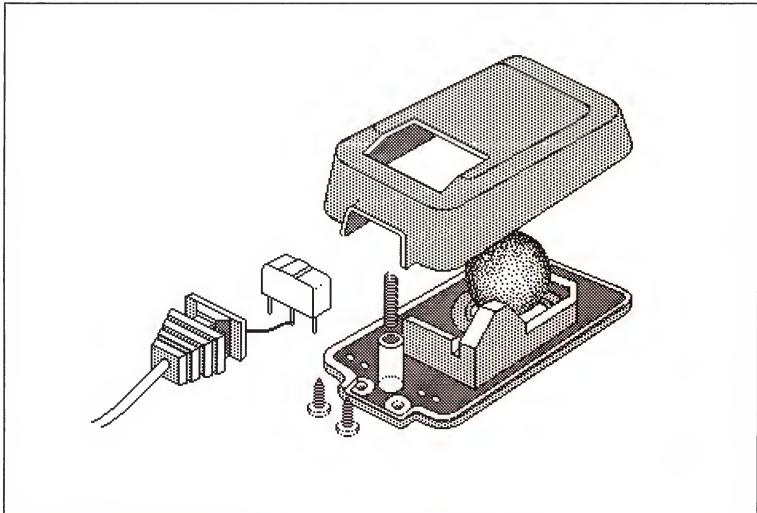
The Desktop Mouse

Most people's first reaction to the mouse or even the idea of a mouse-driven menu system is skepticism; they are leery of the mouse and cannot conceive of a computer without the noble keyboard and the familiar alphabet, numerals, and other symbols engraved on the keystops. No crea-

ture from the lower end of the evolutionary tree (or a mechanized gimmick) can replace the human intelligence embodied in a typewriterlike machine.

But after you spend a few minutes moving the mouse around and performing simple operations on a computer like the Mac, the thought of a standard cursor and the familiar keyboard seems less appealing. At first you may have problems coordinating your hand and eyes as you move the mouse and simultaneously watch the pointer move on the screen. You may end up playing "chicken" with the mouse and the end of your desk as you try to figure out how to reach the topmost cell of a worksheet without sliding the mouse off the desk (just pick up the mouse and try again, and you'll get your bearings). But after a few hours, you'll feel like an expert as you edit a letter or work on your budget with newfound ease and speed.

Using a mouse draws on an innate human skill: pointing. It's as natural as throwing a ball, but more importantly, it increases your productivity and frees you from the constraints of the keyboard. Because the mouse minimizes use of the keyboard, you can concentrate on what's happening on the screen, and you won't have to memorize or look up all sorts of keyboard command codes to do your work. For tasks that don't require extensive typing, a mouse in the hand is an efficient device.



A MacPaint portrait of the Macintosh mouse

Mouse Lingo

As with other practical tools, specific terms are used to describe the actions of a mouse. Initially, you will focus on *sliding* the mouse on the desktop. However, once you get accustomed to working with the mouse, your reference point will be the screen. Rather than concentrating on sliding the mouse, you'll be thinking in terms of selecting or *dragging* objects on the screen; if you *press* and hold down the mouse button when the pointer is over an icon, you can drag the icon to a new location. *Clicking* (pressing and releasing) the mouse button selects a location within a document or activates an object.

Double-clicking (pressing and releasing the mouse button twice in rapid succession) is a shortcut method you'll quickly master; for example, instead of selecting an

icon and then moving the pointer to the File menu to choose a command, you can double-click the mouse button in most instances. This action cuts down on the amount of mouse movement and the number of steps you need to complete certain tasks.

Anatomy of a Mouse

The plastic casing of the mouse hides a rather sophisticated piece of technology. If you look at the bottom of the mouse, you'll see a small rubber-encased steel tracking ball surrounded by a circular ring. When you slide the mouse across a desk, the ball mechanically stimulates two orthogonal shafts that drive signals in two corresponding channels: an *x* channel that monitors horizontal movement, and a *y* channel that monitors vertical movement.

The movement of the mouse is initially detected by optical sensors on each channel. There are two light-emitting

sources and two light-detecting sensors for each channel. An optical wheel at the end of each shaft has slots, or apertures, in the optical signal paths. When the wheel rotates, it breaks and releases the beams of infrared light. The optical signals are in turn converted to electrical *quadrature* signals ($x1$, $x2$, $y1$, $y2$) by means of the sensors. These electrical pulses generate interrupts in the computer. Depending on the status of the *x* and *y* quadrature signals, the interrupt handler will either increment or decrement the *x* and *y* locations in memory.

The computer picks up the speed of the mouse by the number of times the infrared beam is broken or unbroken. The

maximum speed that you can move the mouse without losing pulses (interrupt signals) is ten inches per second. That speed is very quick; however, if you exceed that limit, the pointer may not perfectly track the movement of the mouse.

The Mac allows you to modify certain aspects of the mouse's operation. The Control Panel option in the Apple menu has two mouse-related control features. You can choose from among three settings to determine the length of time between clicks that will effect a double-click.

The other mouse control feature, *scaling*, involves the relationship between the speed of mouse movement and the distance covered by the pointer on the screen. You can set the mouse scaling so that the

pointer addresses each dot on the screen, or you can speed up mouse movement by setting the mouse scaling so that the pointer skips over every other dot on the screen. However, the scaling feature will not work unless you move the mouse rather quickly; moving the mouse slowly will address each dot no matter which option is selected.

Apple's research showed that the one-button configuration makes it easier and less confusing for people learning to use the mouse. For more expert usage, the mouse can be used in conjunction with the keyboard. In *MacPaint*, for example, holding down the Option key while you drag a selected part of a drawing will "stretch" the image.

The mouse is a sturdy beast, but don't let it fall off the table or allow your dog or cat to chew on it. Having your workspace as free of dust as possible helps to keep the mouse in good working condition. You should also clean the mouse periodically. The Macintosh owner's manual gives detailed instructions for this simple procedure.

It's in the Software

The mouse will come of age in the 1980s because the software developers at Apple have created a working environment that superbly implements mouse technology. The mouse did not become a part of the Mac system as an afterthought, but, as in the design for the Lisa, was part of the overall concept from the beginning. The Mac's mouse-driven, uniform working environment allows users to concentrate on their tasks, rather than on memorizing the keyboard commands and eccentricities of various application programs.

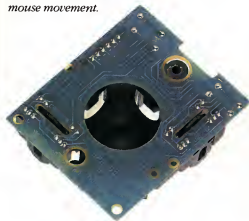
As the community of Mac users multiplies, and as other computer manufacturers adopt similar mouse-based working environments, the mouse will become a constant companion of the keyboard, taking over many of its functions. The mouse may not be the ultimate device for interfacing with computers, but for the time being it's the best system yet devised for making computers more compatible with the people who use them.

■■■■■■■■■■ Daniel Farber is
the Assistant Editor of Macworld.



The Macintosh mouse

The bottom of the mouse. The two rollers within the tracking-ball shell rotate when the mouse (tracking ball) moves, sending signals down two channels that monitor mouse movement.



The tracking ball is easily removed for cleaning the mouse.





Guess what Microtek is doing for the Mac?

Just ask us! Microtek, the company that has been supporting the II+ and IIe for nearly 5 years. After all...

Look what we did for the IIe...

Printer Interfaces

Parallel Printer Card
Serial Printer Card
Graphics Printer Card
Buffered Printer Card

Disk Emulation

128K Disk Emulators
CP/M Disk Emulators
PASCAL Disk Emulators

Video Expansion

80 Column Video Cards for the II+
80 Column Video Cards for the IIe
80 Column w/64K Memory

Color Enhancement

16 Color RGB Drivers
256 Color Programmable
RGB Drivers

Communications

Serial Interfaces for Modems

Memory Expansion

16K Memory Expansion Cards
64-128K Memory Expansion Cards
512K Memory Expansion Systems
with software support

Utilities & Software

Memory Management
VisiCalc Expansion
Color Monitor Programming
CP/M Enhancers
PASCAL Enhancers
Bundled MAGICALC by ArtSci

*For information and to get on our "Mac Mailing" list
Call Toll Free (800) 854-1081 or write to:*



SOFTWARE™ BY
MICROTEK

4750 Viewridge Avenue, San Diego, CA 92123
(619) 569-0900/TLX: 910-335-1269

Congratulations Apple!...Mac!



Introducing MACWORLD

The Macintosh Magazine

**The Exciting New Magazine for
the World's Most Creative Personal
Computer—the *Macintosh***

**At Special Charter Subscriber Rates
Save Up to 54% Off the Newsstand Price***

(*when you subscribe for 3 years)

Personal computing took a giant leap into the future when Apple introduced the Macintosh. We've created *Macworld* as your passport into this incredible new realm of computing. Each issue we'll bring you everything you need to explore and get the most out of your Macintosh. From the latest product news to innovative business applications, from amazing graphics to personal productivity tools and games, each issue of *Macworld* will be packed with interesting, practical, readable information written by skilled, perceptive writers.


Each month we'll be creating a *Macworld* community; sharing ideas, problems, and creative solutions while we explore the world of Macintosh together.

So why take a chance on missing a single exciting issue? For a limited time only you can subscribe to *Macworld* at special Charter Subscriber rates of only

 **\$24 for 1 year (12 issues)**

That's a savings of 20% off the regular \$30 subscription rate and 50% off the newsstand price!

**And you can save even more by subscribing
for 2 or 3 years at Special Charter Rates. Don't
Wait....Subscribe Today and Save!**



ThinkTank™ YOU'LL NEVER LOSE A GOOD IDEA

EVER LOST A GOOD IDEA?

With ThinkTank it's almost impossible. Sit at the keyboard and brainstorm. When an idea comes, put it in your ThinkTank. Relax, then think some more. Another idea comes to mind, then another—but you're prepared.

Elaborate as much as you want on any particular idea. ThinkTank will store it on-screen or "collapse" it into memory off-screen. If, at a later time, you want to edit your "collapsed" information, you can simply "expand" the heading, bringing the detail back into view.

AN ENTIRELY NEW CATEGORY OF COMPUTER SOFTWARE

Idea processing—fragmented thinking, expanding, revising, deleting . . . limitless changing and updating of data; this is how the human mind conceptualizes, creates and stores its refined data.

ThinkTank, available for Apple and IBM personal computers, is a tool which you can use to capture and organize ideas. It adds to your efficiency as a thinker, and helps you refine the presentation of ideas.

YOU NEED FLEXIBILITY

Thanks to ThinkTank, personal computers from coast to coast are helping people get the most out of their ideas. ThinkTank will be your file cabinet, your daily planner, your electronic secretary . . . Flexible, accessible, and constantly updated, ThinkTank is the first IDEA PROCESSOR.

WHAT DO THE PROS THINK?

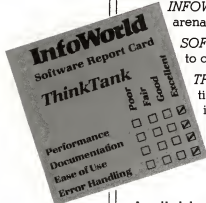
We didn't have to ask. They told us in glowing reviews nationwide.

INFOWORLD (July 25, 1983): "an amazing tool" . . . "your screen becomes a dynamic arena for your ideas."

SOFTALK (August 1983): "get more out of your thinking" . . . "limitless permission to change your mind."

THE NEW YORK TIMES (May 17, 1983): "ThinkTank is so easy to use, and so relatively errorproof that even a first-timer feels as if he's in control of the computer, instead of the other way around."

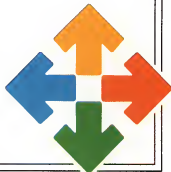
SCIENCE DIGEST (August, 1983): "you may well find yourself hooked."



Available for Apple's Macintosh early in the second quarter of 1984.

See ThinkTank performing on the revolutionary new Macintosh at SOFTCON, Booth A931.

ThinkTank and "the first idea processor" are trademarks of Living Videotext, Inc.
Apple is a registered trademark of Apple Computer, Inc.
IBM and IBM PC are registered trademarks of International Business Machines Corporation.



THINKTANK
The First Idea Processor

Living Videotext Inc., 1000 Elwell Court, Palo Alto, CA 94303 (415) 964-6300

Inside the Mac

A close-up view of the Mac's internal hardware

Matthew Douglas

The Macintosh's hardware is every bit as marvelous as its software. You're probably not surprised. But this fact may surprise you: any competent hardware engineer could have designed the Mac's hardware. Simply take a Motorola 68000 microprocessor, throw in 64K of ROM, add 128K of RAM, slap on a couple of high-speed serial ports, cram in a few additional goodies, and voilà—a Macintosh computer.

Such a computer would have only two flaws: it would be the size of a large window air conditioner and would cost thousands of dollars more than the Mac.

The Art of Design

Good design, whether in buildings, bridges, or computers, is a combination of craft and art. Great design is great art, and great art seldom comes from a team production. One man, Burrell Smith, is primarily responsible for the awesome speed, amazing size, and low cost of the Mac.

Hardware engineers are fond of saying that anything can be designed if you're willing to pay the price of high-cost components and mammoth power supplies. Just divide the engineering problem into sections, and then "throw chips" at each section. Four-voice sound? Throw a big glob of chips on the board. High-resolution monitor? Cram a bunch of chips on a big card and shove it in an expansion slot. Disk drives? Add another big card full of chips. Serial ports for the printers, modems, or networks? More chips, more chips, more chips, and crank up the power supply. The result? Bigger size, greater power consumption, greater cost to manufacture, and bigger price stickers for buyers.

Brilliant engineers, on the other hand, take a different approach. They create designs that squeeze maximum function

from a minimum of chips. The result is small, fast machines that sip few watts, are simple and inexpensive to build, and offer great reliability.

When hardware engineers gather to pore over the Mac's hardware specifications, they are invariably impressed—and with good reason. Until Burrell Smith designed the Mac, no one believed it was possible to fit this much computer onto two small circuit boards stuck into a compact, lightweight case.

The 68000

The heart of the Mac is the digital motherboard, home of the 68000 microprocessor, read-only (ROM) and random-access (RAM) memory, and six amazing chips called PAL (Programmable Array Logic) chips. The motherboard is located horizontally at the bottom of the main unit, in line with the connectors at the rear. An analog board, containing the power supply, speaker, and circuitry needed by the video monitor, is placed vertically along the left, inside edge (when viewed from the front) of the main unit. The monitor tube takes up the inside-middle area, with the Apple/Sony Micro-floppy disk drive tucked underneath.

The motherboard is where the actual processing takes place. At the center of this activity is the Motorola 68000 microprocessor, a 16-bit minicomputer on a single chip. If you could peek inside the 68000, you'd notice that the chip can easily work with 32-bit chunks of information, what Motorola calls a *long word* of data.

By comparison, the Apple IIe, the Apple III, Z-80 computers, and most CP/M machines are 8-bit machines. Their internal unit of information consists of eight binary digits:

00100100

The 68000 thinks nothing of processing data chunks like this:

11001100101011001101010111001010

Processing bigger chunks of information means larger numbers and more complex instructions can be held within the microprocessor and then swapped and manipulated between on-chip data and address registers. The 68000 has 17 on-chip registers to hold addresses, results, and intermediate calculations.

The 68000 communicates with everything else along an address bus 16 bits



The 68000 chip can easily work with 32-bit chunks of information.

wide. Data is slammed in and out 16 bits at a time, unlike the 8-bit maximum for 8-bit microprocessors. The key word is *throughput*; throughput for the 68000 is about double that of 8-bit processors, or 8/16-bit processors crippled by only eight lines to the outside world.

Microprocessors do their work in frequency cycles, similar to the ticks of a metronome. For most 8-bit microprocessors,

*Inside the Macintosh
system unit*

- 1** CRT (cathode ray tube)
- 2** Yoke
- 3** Disk drive
- 4** Analog printed circuit board
- 5** Clock battery
- 6** Power switch
- 7** Power input
- 8** Sound jack
- 9** Serial ports
- 10** External disk drive connector
- 11** Mouse connector
- 12** Digital printed circuit board



the metronome thuds away at 1 or 2 megahertz (MHz). The 8088 processor within the IBM Personal Computer steps along at 4.7 MHz. The 68000 within the Mac sprints at an impressive 8 MHz.

The combination of wide data, a 16-bit bus, 17 registers, and a fast clock speed translates into a chip that processes from one to two million instructions each second. That's some fast thinking.

Memorabilia

The Mac comes to life fairly intelligently, even when the disk drive is empty. That intelligence is contained on two high-density chips that hold 64K of ROM—memory that contains almost 500 separate programs, many of them accessible by other software. The ROM contents are frozen on silicon; they can be used or read by other programs, but can't be written to or changed.

Most microcomputers have between 2K and 32K of ROM. While the Mac's 64K of ROM is impressive, it might be more accu-

as interrupt handlers, device drivers, an asynchronous I/O system, a memory manager, a file system, a segment loader, and various utilities. Other parts of the Mac's ROM are taken up by Bill Atkinson's QuickDraw graphics programs and the Macintosh User Interface Toolbox, a group of "managers" and services intended to help program developers conform to the Mac way of doing applications. The Toolbox includes routines for handling windows, menus, dialogs, text editing, controls, and system resources (see "The 64K Treasure Chest" for an overview of program development on the Mac and the Toolbox units).

The Mac's RAM is comprised of two banks of 8 chips each: 128K of RAM on 16 chips. Like the ROM chips, the RAM chips are "late-model," high-density integrated circuits. Although memory can't be expanded (yet), programs are free to use ROM code. In effect, application programmers will find that the tedious parts of their programs are prewritten for them by Apple's programmers. The attitude of "making it easy for application programmers" should result in Mac programs that look and behave like all other Mac programs.

The mass of ROM programs takes the pressure off the Mac's 128K of RAM. Since the routines to create menus and dialog boxes, handle files, and open windows are already written in ROM, the 128K of RAM can be used for more specific purposes.

Is 128K enough RAM? Of course not—there's no such thing as enough RAM. A Mac with 256K would be better, 512K better still, and even a full megabyte of memory would certainly be fully consumed by RAM-hungry programs, help files, RAM disks, integrated programs, and innumerable crafty utilities.

Expect the 128K limit to be stretched and pulled in a variety of ways: programs that shuttle chunks of code in and out of RAM, files in RAM merged and overlaid on other files, and virtual memory, which fools the Mac into believing that disk space is just so much more RAM.

Another chip, the 6522 Versatile Interface Adapter, handles the Macintosh mouse, keyboard, and real-time clock. The

clock is powered by a pen-light battery accessible from a small panel at the rear of the main unit. Even when the Mac is turned off or unplugged, the battery keeps the time and date constantly updated.

PALs

Possibly the most magical aspects of the Mac's hardware are contained in the six special PAL chips. The PALs achieved a measure of fame for their starring role in Tracy Kidder's *Soul of a New Machine* (Avon, New York, 1982), in which they played a crucial role in the development of a minicomputer.

The PALs also play a crucial role in the Mac. They are the equivalent of entire circuit boards placed on a single chip. Hardware designers determine the programming of a PAL much like they go about designing entire boards of chips. Each PAL takes the place of several discrete components. How many? Again, it depends on the skill of the designer. The 6 PAL chips on the Mac motherboard replace, conservatively, 30 or more conventional integrated circuits.

The PALs work with the 68000 processor, generate different bus and timing signals, and mediate between the motherboard and the video display. It's safe to say that without the PALs and Burrell Smith's inventive PAL programming, the Mac would have had a radically different appearance.

A Graphic Display

Appearances can be deceiving. Most computers display text on one of 24 or 25 "invisible" horizontal lines on the screen. This display is called *text mode*. To display graphics, the software switches to *graphics mode*, and the display becomes a field of dots. Each dot, or pixel, is either off (invisible) or on (visible). Of course, a computer may have more than one text mode or two or more graphics modes, or it may be a mixed mode of graphics and text.

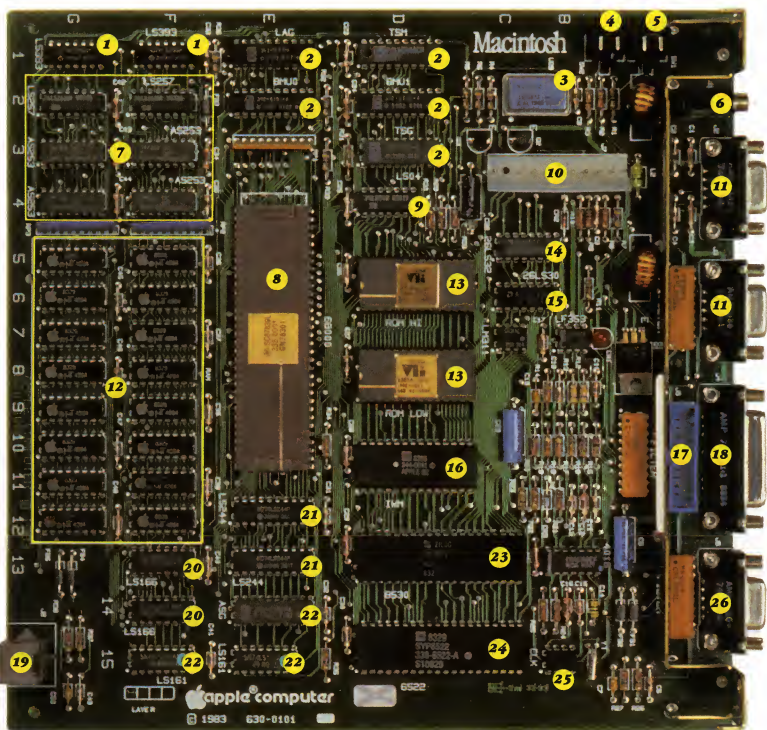
The Mac display has only one mode: graphics. The entire screen is made up of dots: 512 dots horizontally and 342 dots vertically, a total of 175,104 dots that combine to display everything you'll ever see on a Mac screen. (Now you know the secret behind the incredible range of type fonts, attributes, and type sizes.)



While the Mac's 64K of ROM is impressive, it might be more accurate to imagine it as 96K or even 128K of ROM.

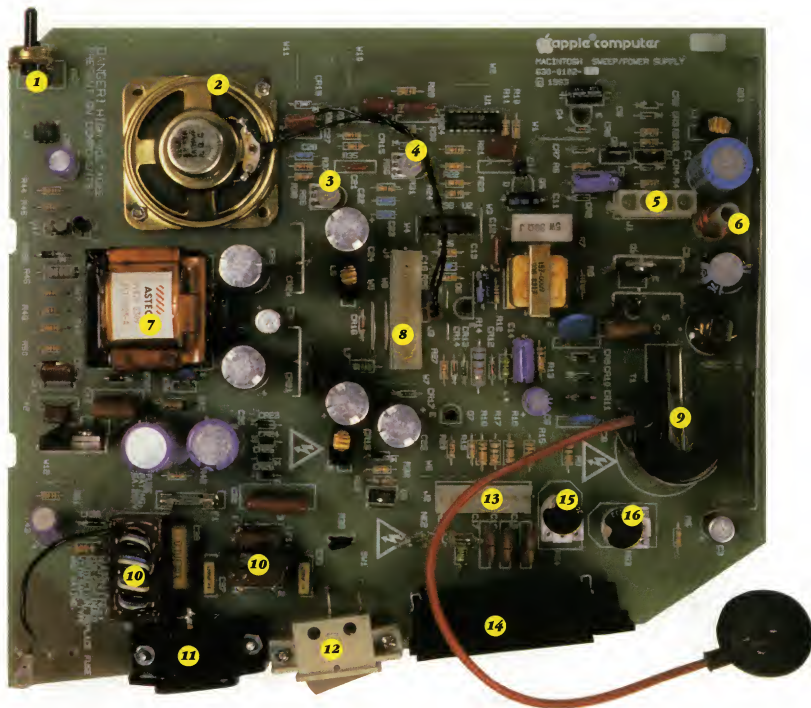
rate to imagine it as 96K or even 128K of ROM. The reason is the incredible optimization of program segments within ROM. Apple's software engineers put enormous effort into making each Mac routine as small and fast as possible, coding, recording, squeezing, polishing, and refining every routine until it was lightning fast and incredibly small. It's the type of work that high-level language compilers aren't smart enough to do; you've got to get down to the gritty level of assembly language and thrash it out with the BSETs, RORs, and SUBQs of assembler.

The ROM chips are a treasure chest for programmers. Inside are some of the traditional operating system components such



***The heart of the Macintosh:
the digital motherboard***

- | | | |
|-------------------------------------------|------------------------------------|--------------------------------------------|
| 1 Video counters | 9 Inverters | 19 Keyboard connector |
| 2 System timing and control (PALs) | 10 Power/video connector | 20 Video shift registers |
| 3 15.6672 MHz oscillator | 11 Serial port | 21 Bus control |
| 4 Reset switch | 12 RAM array (64K × 16) | 22 Sound/disk speed control |
| 5 Interrupt switch | 13 ROM (32K × 16) | 23 Serial communications controller |
| 6 Sound-out jack | 14 Serial receiver | 24 6522 versatile interface adapter |
| 7 RAM address multiplexors | 15 Serial driver | 25 Real-time clock |
| 8 68000 microprocessor | 16 Disk controller | 26 Mouse connector |
| | 17 Internal drive connector | |
| | 18 External drive connector | |



The Macintosh analog circuit board

- | | | |
|---------------------------------------------|----------------------------------|---------------------------|
| 1 User-accessible brightness control | 6 Width coil | 12 Power switch |
| 2 Speaker | 7 Power transformer | 13 Video connector |
| 3 Voltage adjust | 8 Processor connector | 14 Battery holder |
| 4 Height control | 9 Flyback transformer | 15 Focus control |
| 5 Yoke connector | 10 Line filter components | 16 Cut-off control |
| | 11 Power input | |

Serial Speed

The main link between the 68000 and the Mac's two high-speed serial ports is the serial communications controller, the 8530. These serial ports link accessories, such as a printer, to the main unit. The serial ports are side by side on the rear panel; one is marked by an icon representing a telephone receiver, the other by a printer icon. But don't be misled, the ports are identical; you can plug a printer or a modem into either port.

Typically, computers have either a parallel port, a serial port, or one of each. The serial ports are usually termed RS-232C serial ports. The RS-232C designation refers to a communications convention detailing how to send electronic pulses down a line. Unfortunately, RS-232C comes from the days of slow, clunky teletype machines that were linked to slow computers. Imagine blowing big peas through a small straw.

Parallel ports are faster but more complicated. Using a parallel port, you can transmit data in chunks instead of as a

RS-422 hardware can transfer 4 megabytes of information per second—more than enough for typical (and untypical) uses. Four megabytes per second is the hardware limit; software slows the rate down a good deal. Still, it leaves the old RS-232C ports eating dust.

At the top of the soon-to-come list is networking—a Mac on every desk, each with the ability to send messages, mail, programs, and documents to other Macs. When networking arrives—and it won't be long—making the connection will be mostly a matter of plugging a cable into an RS-422 port.

Sony's Contribution

The Mac's 3½-inch micro-floppy disk drive adds another chapter to the speed and storage story. Apple buys its Mac disk drives from Sony. Although 3½-inch disk drives are somewhat new to most buyers, Sony disk drives have been available in computers from Hewlett-Packard (HP), and they've proved to be fast and dependable.

The micro-floppy disks are also sturdy, but they're not floppy, despite the "Micro Floppydisk" label on the package. Each disk is contained in a rigid plastic cover. A spring-loaded shutter opens when the disk is inserted to expose the disk's delicate recording surface. You'll never have to worry about getting fingerprints on the surface of your disks.

Micro-floppy disks are so small that six or seven will fit easily into a shirt pocket. They are sturdy enough to survive being dropped onto the floor, and sail with ease through the hazards inflicted by the U.S. Postal Service. If the Mac is your first computer, you're going to like these disks. If the Mac isn't your first computer, you're going to love them.

The HP disk drives, like those in the Mac, are Sony single-sided drives. The HP/Sony drives store 270,000 bytes on each disk, while the Macintosh/Sony drives have a formatted capacity of 410,000 bytes. The difference is in the hardware. The Mac uses an Apple-designed, variable-speed disk controller. Depending on what disk area is under the read/write drive head, the disk

motor spins the disk at between 400 and 600 revolutions per minute (rpm). The varying speed increases reliability and simplifies the analog read/write electronics. Most 5¼-inch disks, by comparison, poke along at 300 rpm, hold from 80K to 320K of information, and are noisy.

Disk ejection is automatic—no little doors, no fumbling inside the drive. Go to the File menu, drag down to the Eject com-



Macintosh/ Sony drives have a formatted capacity of 410,000 bytes.

mand, and release the mouse button. A good two inches of the disk pops out of the slot.

"But what if I can't make the software give me the disk?" you ask. Apple thought of that. In the absolute worst case imaginable, you can switch off the Mac, hold down the mouse button, and switch on the machine again. This procedure always ejects the disk.

For additional storage, an external disk drive can be plugged into a connector alongside the serial ports. Currently, Apple supports only one additional disk drive. The operating system software, however, supports up to four disk drives and will also support double-sided disk drives when they become available. Double-sided drives would allow not 410K, but 800,000 bytes per disk.

Analog Bored

Analog electronics are often overlooked. They're just not as glamorous as high-speed microprocessors, fast RAMs, or snazzy disk drives. But they're important nonetheless. The Mac's analog circuit board contains the power supply, power



The Mac display has only one mode: graphics.

stream of individual bits. Until now, you needed a college degree (or an exceptional salesperson) to determine the proper serial or parallel printer that would work with the proper serial or parallel port and the proper serial or parallel card. Don't forget trying to find the right cable.

The Mac does away with the confusion by adopting a new serial convention: RS-422. The beauty of the RS-422 is fourfold: it is fairly uncomplicated; it has a high bandwidth (which, loosely speaking, means it can pack lots of information, including voice, into a cable); it is the new national standard for serial communications, meaning that other computers will eventually have RS-422 ports; and it is fast.

Just how fast is difficult to pinpoint. In most cases, the serial port handles data faster than the computer can process it, which is very fast indeed. Roughly, the

Computer Jargon

Janet McCandless

You don't need any technical knowledge to operate the Macintosh. You can design complicated electronic spreadsheets, write memos and letters, or create an elaborate drawing without being bothered by the computer's *bits*, *bytes*, *RAM*, or *ROM*. However, if you want to know how the Mac is able to perform your computing tasks, you'll need a basic understanding of the technical jargon. The following glossary briefly explains some of the terms referred to in "Inside the Mac."

- **Address bus:** a transport and identification system whereby circuits route (bus) information from a memory location where information is stored (an address) to another location.

- **Analogue circuit:** a circuit that handles information as a continuous range of physical values such as voltage, light, or temperature.

- **Bit:** a contraction of *binary digit*. The computer identifies each unit of information as on or off voltage in a circuit. Bits are grouped together to form larger units of information such as a nibble (4 bits) or a byte (8 bits).

- **Byte:** an 8-bit unit of information that represents one character of data.

- **Chip:** an integrated circuit formed by thousands of electronic components etched onto the surface of a silicon wafer.

- **Digital circuit:** a circuit that handles information as distinct, separate values, most frequently as binary values of on or off, 1 or 0.

- **Disk drive:** a device that rotates flat, circular magnetic storage media (disks) past an electromagnet (head) so that information can be recorded, retrieved, or erased.

- **K (kilobyte):** a measure of computer memory; one K equals 1024 bytes, or approximately one-half typewritten, double-spaced page of text.

- **Microprocessor:** an integrated circuit usually contained on a single silicon chip that is the central processing unit (CPU), or "brain," of a microcomputer. The microprocessor performs arithmetic and logical operations and obtains and decodes instructions.

- **Modem:** a device that links computers into a communications network. Information is translated into signals that can be transmitted from one computer to another by telephone, satellite, or other switching systems.

- **Motherboard:** the main circuit board onto which smaller circuit boards or chips are mounted.

- **Network:** a means of interconnecting computers so that information can be easily exchanged and peripheral devices shared.

- **PAL (Programmable Array Logic):** an integrated circuit whose exact logical function is programmable at the time of manufacture.

- **Parallel port:** an outlet (port) that connects the computer with a printer or other peripheral device through which several bits of information are received or transmitted at once. Parallel transmission works faster than serial transmission since more information can be sent or received at a time.

- **RAM (random-access memory):** the memory available to the user for storing information or instructions. RAM is read-write memory; it lets you retrieve (read) information you have put in the computer or add (write) new information. Data stored in RAM will be lost when the computer is turned off unless it is saved on a magnetic medium such as tape or disk.

- **Register:** a temporary storage unit for information currently being processed by a computer such as arithmetic or logical operations.

- **ROM (read-only memory):** the memory the computer uses to store its operating instructions. The information in ROM is unalterable and remains in the computer even when you turn it off.

- **Serial port:** an outlet (port) that connects the computer with a printer or other peripheral device through which information is received or transmitted one bit at a time.

- **Throughput:** a measure of productivity based on the number of instructions executed.

supply regulator, vertical and horizontal scanning circuitry, other video features, and the Mac's speaker.

Power supplies tend to be heavy and expensive. The trick to designing a good power supply, according to Apple's engineers, is to be "as clever and smart as we can be and still be flexible." Translated into circuits that means an extremely lightweight switching power supply capable of delivering clean power, even when the power from the wall socket is far from optimum. The Mac's power supply can handle surges and subnormal amounts of power (within reason), and even keep the Mac running smoothly during momentary breaks in power.

If your house has old wiring (or if you're subjected to a particularly poor power company), you might notice that the lamps in your house sometimes dim momentarily. If so, don't worry; your Mac won't flicker, dim, or skip a beat.

Three watts from the power supply are reserved for the internal speaker; enough power to get a respectable volume

from the 2-inch speaker. How much volume is up to you. Unlike other computers, the Mac allows you to vary pitch and volume and to play up to four notes at the same time.

The prospect of four-voice chords might make you wish for more volume. Apple thought you might. For better tone,



*The Mac is the
most sophisticated
computer ever offered
in its price range.*

plug a 4 ohm to 10 ohm external speaker into the audio out jack on the rear panel. For "eviction notice" sound levels, connect the Mac to the proper plug on your stereo. The Mac then becomes, in effect, a pre-amplifier; the volumes that result will leave you happily dazed.

The Big Win

The Mac might fool some people. Because it's small, some may think it's a toy. Because it's inexpensive, some may think it can't possibly be useful for business applications. Because the monitor is black and white, some may think that graphics will suffer.

A close look at the Mac's hardware proves otherwise. The Mac is the most sophisticated computer ever offered in its price range—and faster and more powerful than many machines that cost eight to ten times more.

The Mac has been the subject of fevered speculation in the past few months. The consensus was that Apple was about to win big or lose big. It's still too early to tell how the Apple marketers will fare or what the competition will be, but one thing is clear: from a hardware viewpoint the Mac is truly a big win—rugged, fast, powerful, expandable, and affordable. A thoroughly marvelous design.

Macintosh Specifications

Processor

- Motorola MC68000, 32-bit architecture, 7.8336 MHz clock frequency

Memory

- 128K bytes RAM
- 64K bytes ROM

Disk capacity

- 400K bytes per formatted disk, 3½-inch diameter hard shell media

Screen

- 9-inch diagonal high-resolution black-and-white 512-by-342-pixel bit-mapped display

Interfaces

- synchronous serial keyboard bus
- two RS-232C/RS-422 serial ports, 230.4K baud maximum

(up to 0.920 megabit per second if clocked externally)

- mouse interface
- external disk interface

Sound generator

- 4-voice sound with 8-bit digital-analog conversion using 22 KHz sample rate

Clock/calendar

- CMOS custom chip with 4.5 volts user-replaceable battery backup (Eveready No. 523 or equivalent)

Input

- line voltage: 105 to 125 volts AC, RMS
- frequency: 50 or 60 Hz
- power: 60 watts
- keyboard: 58-key, 2-key rollover, software mapped
- mouse: mechanical tracking, optical shaft encoding, 3.54 pulse per mm. (90 pulse per inch) of travel

Size

- main unit: 9.7W × 10.9D × 13.5H inches
- keyboard: 13.2W × 5.8D × 2.6H inches
- mouse: 2.4W × 4.3D × 1.5H inches

Weight

- main unit: 16 lbs. 8 oz.
- keyboard: 2 lbs. 8.5 oz.
- mouse: 7 oz.

Environment

- operating temperature: 50°F to 104°F
- storing temperature: -104°F to 122°F
- humidity: 5% to 90% relative humidity
- altitude: 0-15,000 feet



Polishing the Mac

*An interview
with Microsoft's Bill Gates*

Edited by David Bunnell

Bill Gates, Chairman of the Board of Microsoft, first learned of the Macintosh from Steve Jobs in June 1981. By January 1982, a full two years before the machine's introduction, Gates and a team of programmers were writing Mac software and participating in the Mac's development. Their goal was to have a range of software tools including Microsoft BASIC, *Multiplan*, and other application packages ready for the Mac's launch or soon thereafter.

Microsoft's roots in the computer industry go deep. The company has made major contributions to the success of many of personal computing's most popular machines. Most notable has been the operating system for the IBM Personal Computer (MS-DOS, aka PC-DOS), which has become the standard for 16-bit personal computers. Today Microsoft software is used on over five million personal computers worldwide.

Because of his unique experience, Gates is ideally positioned to understand both the history and the future development of personal computing. In this interview, *Macworld* Publisher David Bunnell learns why Gates has already declared the Macintosh to be a "classic."

Bunnell: When I interviewed you for PC Magazine's inaugural issue in January 1982, you said the IBM Personal Computer was not a machine you would buy for your mother. How do you feel about the Macintosh?

Gates: The Macintosh is far easier to use than anything we've seen before, especially because of what it's letting the software do. Although some of the Mac's

techniques and features can be found on earlier machines, the Mac heralds a major change in how people view and interact with application programs. That's why I'm so excited about it. There's no question that I'll let my mom try it out.

Bunnell: Why can't you do the same things on the PC? What is so special about the Macintosh?

Gates: The Mac was designed as a graphics machine. Apple didn't put in a ROM character generator or a bunch of video modes. They put in only one video mode, and that's the pure bit-mapped, 512-by-342-pixel screen. The monitor was designed into the machine so that they could get extremely crisp pictures and have one integrated system. They knew what the aspect ratio was and how the dots would appear. And they also made sure that the mouse would be used and that the 64K ROM would support very rich graphics interaction.

You can configure a PC with one of the better graphics boards and add a Microsoft mouse and the necessary software, but that's not the thrust of the machine. The PC is used primarily in its text mode, and to date it's used mostly without a mouse; you couldn't get performance or graphics like the Mac's out of the PC at a comparable price. Although they're both "turing" machines (that is, they have finite memory), the thrust of the Mac is quite different.

Of all the personal computers available today, the Mac is unique. It's the first time somebody said, "We don't need a lot of the things that other personal computers have, so let's optimize a few areas and make sure the software is designed around them."

Burnell: When did you first become aware of the Macintosh?

Gates: I talked to Steve Jobs about the Macintosh project in June 1981. We were excited about developing new software and somewhat disappointed that we weren't more involved in the Lisa's development. We thought Apple ought to do a cheaper version of the Lisa, and we didn't think the machine had to have so much memory. We also thought we could move some of our software over, and we'd always wanted to have a machine with a straight bit map.

Steve offered to show us what he was doing on a confidential basis. We went to Cupertino in October 1981 and saw a running prototype. It was running a simple bouncing-ball program, but even so it was impressive, and they were talking about a very aggressive price. We actually signed a contract to work together in January 1982, and agreed to devote a lot of good people to developing a full range of productivity tools for the Mac.

Burnell: Can you tell us something about the nature of your involvement? Is it just creating application programs?

Gates: No, we've had a very in-depth involvement. Whenever you get involved with prototype machines that early on, you are essentially part of the engineering team; you're helping to find bugs and making suggestions for design changes. Microsoft has been an extension of the internal Mac software team for the last few years. We've had a close working relationship with the Mac team that has been beneficial for both sides. We've learned a great deal about doing graphics applications, and we've made sure that their subsystem, dialog boxes, and memory manager fit in properly.

If you compared the Mac to what it was two years ago, you probably wouldn't recognize it. Steve's vision of where the machine should go—that it should be a simple, inexpensive graphics machine—has been preserved. But the disk, the memory, the code in ROM, the number of bits on the screen—they're all different. We didn't realize that we needed to do so much work with the memory manager, menus, and dialog boxes. Nor did we know how we were going to make the Finder work or how the desktop tools would



Bill Gates discusses Microsoft's role in developing the Mac with Macworld Publisher David Burnell

work. All of these things eventually got built into the software that Apple includes with the system.

Microsoft's *Multiplan* was running over a year and a half ago on the Mac. You could do recalcs, resize the sheet by moving the mouse around, and other things. We thought we'd be finished with that product in a few months, and in fact everyone's schedules were overly optimistic. But the product has evolved since then. We started to use the graphics much more, changed the way scroll bars work, and altered the way data is passed between applications. We also sped it up considerably and made sure that the worksheet size would be adequate.

Look at *MacPaint*, the program that Bill Atkinson did internally at Apple. It was impressive a year and a half ago, but it wasn't even close to what it is today. It's been polished up since then. For example, the spray can was included about nine months ago. Every time we'd meet with Apple, we'd show them our new stuff, and they'd show us theirs. We'd tell them that if we wanted to do something even better, we'd need more support in the ROM.

As with any ROM, you keep trying to freeze the thing, because it's got to be extremely reliable and there's a size limit. You also have to consider the lead time for

masking. Four or five times Apple said to us, "OK, this is the final ROM."

Burnell: You mentioned the memory size. How do you feel about the whole memory issue—is it enough for your applications?

Gates: It's certainly enough memory. The Mac started out with 64K, which is one-sixteenth of what the Lisa started out with. Because the Mac's bit map is smaller than the Lisa's, we thought we could do something with that amount of memory. But we were pushing for 128K all the way, and about a year ago we switched to 128K. We figured out how to squeeze the applications down to that size.

When you're writing applications that are going to be simple to use, it's important to have some boundaries that prevent you from throwing in an unlimited number of features; the memory size provides that limit. Certainly what we've got in terms of *Multichart*, *Multifile*, *Multiplan*, and Microsoft BASIC on the Mac are as rich as on any other machine we've seen. I think the people at Apple would openly admit that *Plan*, *File*, and *Chart* are more powerful than their equivalents on the Lisa, and yet they run on an eighth as much memory.

When you do get more memory, you'll be able to have multiple applications active or have more data space available. It's partly these boundaries that have forced us to find more clever ways to do things and stay within the memory size. It's caused us to be more innovative than we would have been if we'd had a megabyte.

Bunnell: *What programming techniques do you use to work within the memory limit?*

Gates: You've got to take advantage of what's in the ROM and learn to do very tight code. It's not simple to write software for the Mac. Over time, as really good program development tools emerge, people will become proficient at writing programs for the Mac.

Look at the Apple II. It took about two years before people were really exploiting its capabilities. If you look at the IBM PC, products such as *Microsoft Word*, *Flight Simulator*, or *1-2-3* took a year and a half to emerge. That's because you've got to get the tools together—you have to understand how to push the machine to its limit. I think we started that cycle much earlier with the Mac. Maybe a year and a half from now I'll say, "Wow, even I didn't realize the Mac could do this newer stuff!" For example, some facilities are available for music in the Mac that we're not using well yet. And there's even talk about how that capability could be extended to include voice synthesis. In the peripherals area, the Mac has those two RS-422 ports, and nobody's really exploiting them yet.

Bunnell: *What impact do you think the Macintosh will have on personal computing?*

Gates: I think the Mac will mean that there's at least one company besides IBM in the personal computer business that doesn't have to do everything the IBM way. It's healthy for the industry to have Apple offering an alternative. The Apple II and the IBM PC will be joined by a third fantastically successful machine—the Mac—so we're expecting to develop a lot of software for it. It's more than just the Mac's hardware and software that's being approached in an innovative way—some innovative work is being done in terms of getting sales people and college students involved with this machine.

The reason people like the computer business is that it's innovative. We've got to continue to experiment with how computers are built and come up with new ideas, or else this industry isn't going to stay as exciting and fast moving as it is now. I've always said that the Mac is an ideal machine for office use, but it's also a great general purpose machine. We think it will address several markets and remind people that the pace of innovation hasn't slowed down in the industry.

Bunnell: *Do you think the Mac is going to change the way people use computers?*

Gates: We've always promised people that they could simply buy a computer, turn it on, and use it. Even the Mac doesn't fulfill



Microsoft has been an extension of the internal Mac software team for the last few years.

this promise, but it's a new milestone in terms of usability. The way the menus and graphics work is so compelling that you're enticed to learn that next feature—you want to try it out and see if it works the way you think it should. The Mac has so little hidden behavior that it will attract a new class of users.

Many people have held back, thinking that computers are complicated, mathematical, scientific-type machines; they haven't viewed them as a medium of expression, as machines with which you can do creative things. But many people will sit down and use *MacPaint* or *Multibart* and realize that the Mac is not confined to the computer stereotype. I'll enjoy having some people I know who are new to computers play with the Mac and see if it doesn't draw them into the excitement of interactive personal computing.

Bunnell: *Do you plan to have programs such as Flight Simulator on the Mac?*

Gates: We're working on many products, and *Flight Simulator* is certainly a good product. Bruce Artwick originally did *Flight Simulator* for the Apple II. He pushed the Apple II to its limit and proved that the machine could do many things that people didn't think it could do. When the IBM PC came out, Bruce proved that the PC could do many things that people didn't think were possible. Now he's taken on the challenge of proving to all of us that the Mac can do things that we don't think are possible. He's talked about things like shading and surfaces, and about getting more realism into the program than what's in the PC version. Certainly within the next year he'll generate another fantastic program.

Bunnell: *What about the lack of color? You have all of these beautiful graphics but no color.*

Gates: The perfect machine will never exist, and the obvious difference between the perfect machine and the Mac is that you could ask for more memory, color, and a faster way to move data in and out of the machine. But when you look at price, you'll acknowledge that this machine is incredible. The Mac is an 8 MHz 68000 machine; if you take a PCjr, configure it up, and set it side by side with the Mac, there will be more than a three-to-one difference in raw horsepower. Since the PCjr has about two-thirds the horsepower of a PC, the Mac has easily twice that of a PC. That's a lot of power, and that power can be exploited in many ways, including moving bits around on the screen.

If Apple had tried to put color in this machine, it wouldn't have been available for a few years, because we'd need three times as many bits for the screen and more memory. We'd have to use 256K chips, and the design of our applications would be different. I think Apple made the right trade-off, but someday a Mac or the equivalent of a Mac with color would be wonderful.

Bunnell: *Some people have predicted that the Macintosh is going to be a classic. Do you think that's correct?*

Gates: The Mac is fun and exciting, and it fulfills Apple's image of producing very personal, interactive, state-of-the-art computers. It's a blend of Apple's knowledge of what personal computing should be and what the semiconductor industry has made possible. It's a great mixture, and it's something that Apple is perfectly positioned to sell to people. The Mac will be remembered as one of the great classic machines.



Introducing MACWORLD

The Macintosh Magazine

**The Exciting New Magazine for
the World's Most Creative Personal
Computer—the *Macintosh***

**At Special Charter Subscriber Rates
Save Up to 54% Off the Newsstand Price***

(*when you subscribe for 3 years)

Personal computing took a giant leap into the future when Apple introduced the Macintosh. We've created *Macworld* as your passport into this incredible new realm of computing. Each issue we'll bring you everything you need to explore and get the most out of your Macintosh. From the latest product news to innovative business applications, from amazing graphics to personal productivity tools and games, each issue of *Macworld* will be packed with interesting, practical, readable information written by skilled, perceptive writers.

Each month we'll be creating a *Macworld* community; sharing ideas, problems, and creative solutions while we explore the world of Macintosh together.

So why take a chance on missing a single exciting issue? For a limited time only you can subscribe to *Macworld* at special Charter Subscriber rates of only

 **\$24 for 1 year (12 issues)**

That's a savings of 20% off the regular \$30 subscription rate and 50% off the newsstand price!

**And you can save even more by subscribing
for 2 or 3 years at Special Charter Rates. Don't
Wait...Subscribe Today and Save!**



**"GET SOPHISTICATED MAINFRAME
COMMUNICATIONS FOR MY MACINTOSH—
EVEN THE CAPABILITY TO SEND,
RECEIVE AND PROCESS DATA FILES?"**

"YOU CAN WITH PERSYST."

Persyst communications products are already proven on the IBM PC*. Now they bring powerful new capability to your Macintosh*.

With Mac/3270* and Mac/3770*, you can link your personal computer to mainframe computers. Plus receive, transmit and process data files.

For sophisticated communications and expansion products for the Macintosh, insist on Persyst.

PERSYST™

Persyst Products, Personal Systems Technology, Inc.
15801 Rockfield Blvd., Suite A, Irvine, CA 92714.
Telephone: (714) 859-8871. Telex: 467864

Sams Presents **INTRODUCING** **THE APPLE®** **MACINTOSH.™**

The Macintosh is here.
And so is Sams book.
INTRODUCING THE APPLE
MACINTOSH, No. 22361, \$12.95.

It's the first and only complete Macintosh book that explains everything about the system—how it works, what software it uses, and more.
Now available at any Sams dealer.

SAMS™

What Technology Is All About.

Howard W. Sams & Co., Inc.
4300 West 62nd Street
P.O. Box 7092
Indianapolis, IN 46206
317-298-5566 or 800-428-SAMS

Apple is a registered trademark of Apple Computer, Inc.
Macintosh is a trademark of Apple Computer, Inc.

ODESTA HELIX

An Interactive Information System
For Individuals With A Need To Know

Odesta Helix is a new stage in the evolution of information/knowledge systems. **Odesta Helix** is an interactive environment. It accepts information that has or may have importance to you, without formatting constraints, and without restrictions of size and complexity. It allows you to transform information creatively, according to your own style and approach—ordering, querying, testing, combining, and recombining. Questions find answers. Problems find solutions. **Odesta Helix** provides an extension of your own thinking. Uniquely, **Odesta Helix** does not dictate process, impose arbitrary order, or get in your way. Instead of obstructing your path, it opens new paths. **Odesta Helix** incorporates windows and icons, so that you can see the relationships between things and ideas, and synthesize new forms. Through its "intelligence", **Odesta Helix** will even inform you whether conditions you specify have or have not been met—in your inventory, control group, test market or cash-flow. You may organize a family's finances, plan field work, or prepare a business plan. You may coordinate, correlate, and quantify research, or compare and qualify opportunities. You may posit and test a hypothesis, or test for significant differences. You may alter schedules to project staffing requirements, or formulate criteria for an evaluation project. **Odesta Helix** is designed for the **Macintosh**®, **Lisa**®, and **Apple**®//+ computers, so that your employees, students, and co-workers can use the form-models you create to add to or exchange information—no matter which machines they are working with. Because of this versatility, you can continue your work as you move from office to site to home to lab. And for the professional, business, academic, artistic and scientific communities, experts are now developing resource models for **Odesta Helix**. They can be used as complete applications or as easily customized templates to work with your own generated applications. Finally, **Odesta Helix** is as easy to use as it is difficult to describe.

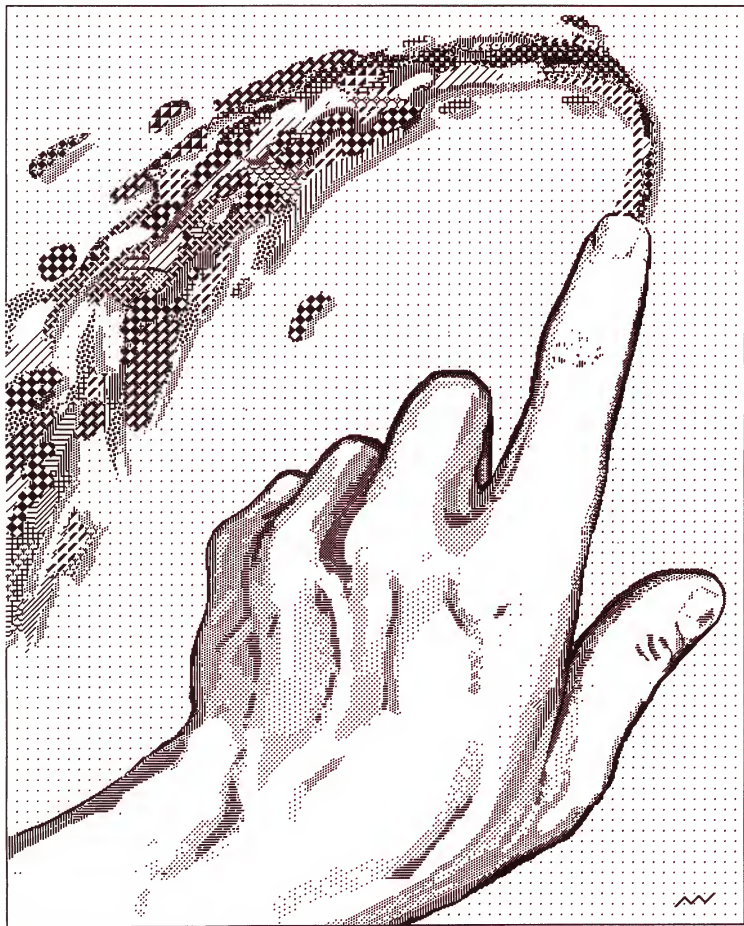
MacPaint: the Electronic Easel

Jeffrey S. Young

Imagine being an artist. Not just any artist, mind you. Imagine being a computer artist, with the kind of processing power computers have long applied to words now extended to drawing and painting. Imagine that you can reach up onto a tool shelf, take down one of 32 different paintbrushes, dip it into one of 38 patterns arrayed on your electronic palette, and with a sweep of your hand produce complex brushstrokes in crosshatches and herringbones, zigzags and pinstripes, rattans and checkerboards. Maybe you want to mirror your strokes as you draw them on up to four axes. Or perhaps you want to flip the result horizontally or vertically, or rotate it by 90-degree turns, or even invert the blacks and whites.

Then you need to clean up a couple of ragged corners in your drawing, so you zoom in and tidy up your image at the dot-by-dot level, moving the page around as though you were sliding your drawing back and forth under a microscope.

Perhaps as you look at your entire design you decide there's only one section that you like, and the rest is dispensable. You select the section you want, copy it, and paste that copy into a document. Your original is intact, the new document has the detail you prefer, and you've done it in moments.



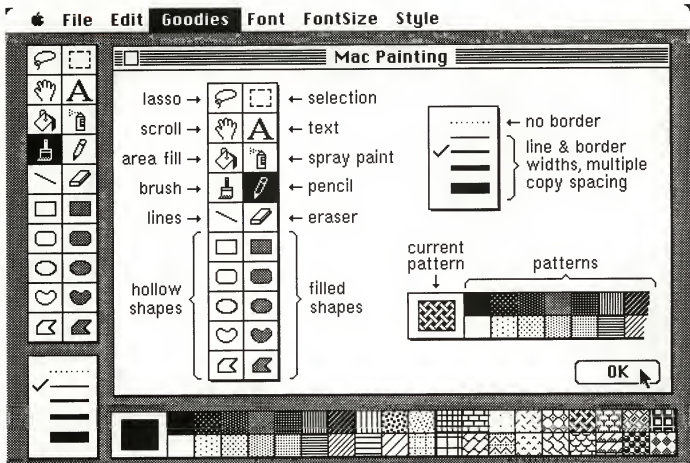


Figure 1
The Introduction option from the Goodies menu provides a quick reference to the MacPaint features.

MacPaint is the electronic equivalent of having top-flight artist's tools and skills at your fingertips.

Save the new copy, return to the original file, and with a double-click on the eraser the screen you started with is wiped clean. No tiny eraser rubbings covering your desktop, no glue stuck to your fingers, no shreds of trimmed paper blowing across the hardwood floors, no smudged thumbprints and crinkled originals. And the drawings you create needn't exist only on the screen. Anything that appears on the screen can be printed with remarkable accuracy.

And that's only the beginning. *MacPaint*, the Macintosh's native graphics program, is the electronic equivalent of having top-flight artist's tools and skills at your fingertips—except the artistic talent that separated most of us from the arts at an early age is no longer as relevant. The limitless possibilities of a blank canvas are fully equaled by the blank screen confronting you as the program starts up... perhaps more than equaled with the diversity of patterns and tool choices available on the palettes surrounding the electronic canvas. The only elements missing are the smell of linseed oil and the drips of paint on your clothes.

A Brief Introduction

From the moment you double-click on the *MacPaint* file in the disk window and watch as the blank canvas appears, followed quickly by the tool, border, and pattern palettes, you know that you've stepped into a different world. If you are overwhelmed by all of the choices that have suddenly appeared on your screen, a good place to start is with the Introduction in the Goodies menu. This menu option displays a screen with the *MacPaint* tools and palettes labeled (see Figure 1). Although most of the icons are self-explanatory, this feature should clear up any questions.

The Paintbrush and the Pencil

Tools are at the heart of *MacPaint*, and the most important of these are the paintbrush and the pencil. You select the tool and pattern you want by clicking the pointer on the appropriate icon. The *paintbrush* can paint in any of the patterns displayed in the pattern palette at the bottom of the screen, using shapes available from the Brush Shape option in the Goodies menu (see Figure 2). You can also look at the brush shapes by double-clicking on the brush icon. The *pencil* draws a narrow, black, freeform line (clicking the mouse button changes the pencil color to white when you are drawing on top of filled patterns).

You may have some trouble getting used to these tools. It took me quite a while to master the delicate interrelationship between holding down the mouse button and moving the mouse (the combination of actions needed to actually put something on the screen) and the resulting image on the page. Drawing is a delicate operation usually performed by a utensil held in the fingers, not a small plastic box with a button on top held in the hand. Perhaps someone will design a mouse the size and shape of a ball-point pen, making it even easier to draw on the screen.

As a freeform tool, the pencil is much more difficult to master than the paintbrush. Because the various brush shapes are wider than the single-dot pencil, making a less precise brushstroke is hardly noticeable, whereas the pencil creates unappealing, ragged lines (see Figure 3). However, once I had learned to use the Shift key in conjunction with the pencil, I was able to create straight, smooth, and precise lines with no hint of roughness. Unfortunately, using the Shift key allows only vertical or horizontal lines to be drawn; with freeform drawing (in which the Shift key option is unavail-

able) you still have the problem of jagged lines. The *straight line* tool also provides a rapid means of producing straight lines in any configuration, in any of the four widths shown in the border palette in the lower-left corner.

Shapes

The ten filled and hollow geometric shapes in the tool palette provide a wide variety of possibilities for drawing. The hollow shapes display a black outline in a width determined by the border palette selection, and leave the interior either empty or transparent to the design behind them. The solid shapes also display the outline in the currently selected border width, while filling the interior of the shape with the currently selected pattern. If the dotted line in the border palette is selected, the filled shapes will have no outlines.

The *rectangle*, *rounded-corner rectangle*, and *circle* take on continuously changing shapes as you move them with the mouse (see Figures 4 and 5). Using the Shift key with the mouse, however, provides perfect squares and circles.

The *freeform shape* and the *polygon* let you create shapes in any combination or configuration. You can fill them in with patterns or vary the border width just as with the predefined shapes. Any enclosed object, either straight-line or freeform, can be filled with any pattern.

The Paint Bucket and the Spray Can

The *paint bucket* fills in enclosed spaces, providing a quick way to insert broad swatches of patterns throughout a drawing (see Figure 6). When using the paint bucket, you must make sure that the section you want to fill with a pattern is actually an enclosed space. If even a single dot is missing from the perimeter, the paint bucket will spill the pattern across your drawing until it finds an enclosed section somewhere else; if no other sections are enclosed, it will fill the entire screen. Don't worry if this happens; you can mop up the spilled paint by selecting the Undo command from the Edit menu.

In the days before *MacPaint*, filling an area with a pattern would have taken hours to accomplish. Now it can be done in seconds using any of the choices in the pattern palette.

MacPaint takes the spray can out of the street and makes it into a nimble artistic tool. The *spray can* sprays any pattern selected in a realistic way (see Figure 7). With the first few passes, the filling-in is faint, as with a real spray can; if you continue to cover the same area, it fills in gradually until you have a solid belt of paint. Perhaps the Mac will bring graffiti into the computer age; you'll turn on your computer one day to discover a rude comment sprayed across one of your files.



The Eraser

The ability to erase is an excellent example of the flexibility of *MacPaint*'s program design. The two primary ways to erase are selecting the *eraser* and dragging it over the part of the drawing you want to remove, or double-clicking the mouse on the eraser icon, which clears the entire drawing screen. When you click on the eraser icon, a box about a quarter inch square appears on the screen (see Figure 8). This shape is quite unwieldy if you need to erase a small section of a detailed drawing.

An efficient way to erase is to "paint with white." Choose the white pattern from the pattern palette and the paintbrush from the tool palette, select a brush shape, and then paint over the portions of the drawing that you want to erase. The advantage of this technique, although it is a little more complicated than simply selecting the eraser, is that the wide choice of brush shapes allows you to be much more precise in what you remove.

FatBits

One of the great qualities of this program is that any drawing you create on the screen can be cleaned up at the single-dot level. So what if the line you drew with the pencil is a bit jagged? All you have to do to improve details of your work is select the FatBits feature, either by pulling down the Goodies menu and choosing the FatBits option or by double-clicking on the pencil icon. An enlarged dot-by-dot pattern of a section of your drawing (determined by the location of the last screen activity) is displayed, as well as an overall picture of the entire drawing in the upper-left corner of the screen (see Figure 9). In FatBits, the pencil can either insert or erase a dot, and you can watch the success (or failure) of your efforts in the insert screen in the corner. When you're satisfied, a single click in the insert screen returns you to the full drawing.

This is a remarkable capability. FatBits enables you to zoom into the 99,840 dots that make up the *MacPaint* screen and manipulate each individual dot. This ability to work on details, coupled with the virtually inexhaustible supply of scrap paper that the eraser gives you, allows anyone, with or without any artistic skills, to turn out respectable drawings with minimal practice.

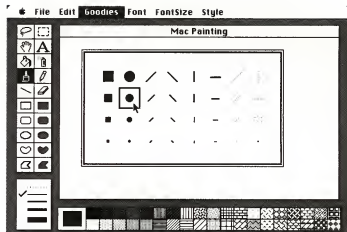


Figure 2
You can choose from among the 32 paintbrush shapes shown in the Brush Shape option.

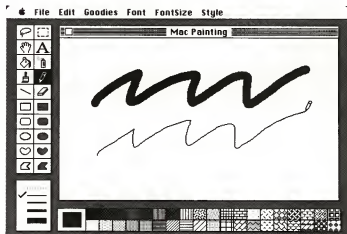


Figure 3
Paintbrush strokes are more attractive and less difficult to create than freeform lines drawn with the pencil.

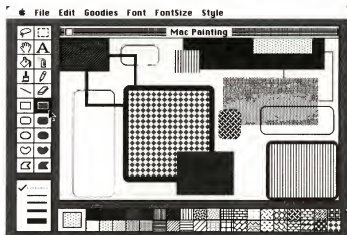


Figure 4
Filled and hollow rectangles can be enhanced using any of the patterns and the various border widths.

Creating Patterns

One of *MacPaint*'s unique capabilities is revealed when you drag the pointer down onto the pattern palette and double-click on any of the 38 choices. The result is a pattern edit window in which you can modify the dot-by-dot representation of each pattern while watching the effect of your changes on a full-pattern representation next to it (see Figure 10). With a little practice (it's not that easy to get used to the limited 8-by-8-dot pattern easel), this feature enables you to create an entire series of customized patterns.

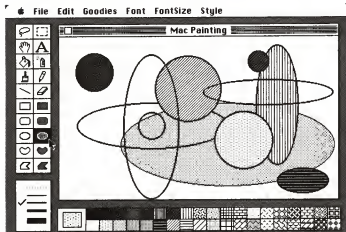


Figure 5
Circles and ovals can be filled with any pattern and outlined by any of the border widths.

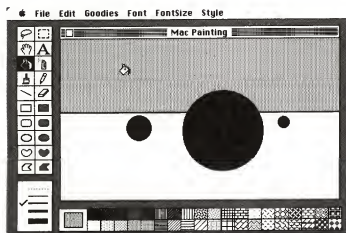


Figure 6
The paint bucket fills enclosed spaces with the currently selected pattern.

Once you've changed a pattern to your liking, click the box labeled "OK." The Mac will store it in the pattern palette of the current *MacPaint* file. Whenever you return to that file, the patterns you created will be available. The default set of patterns will always appear on a new file created from the master *MacPaint* program, however. Having the ability to customize the program is one of the most appealing qualities of *MacPaint*.

Selectors

To fully utilize the program, you must become familiar with the two selector tools, the *lasso* and the *selection rectangle*, or *marquee*. They provide ways of selecting a part of the drawing on the screen and performing any of a series of operations on it (for instance, all of the Edit options require a selected piece of the drawing to operate on).

These two tools are quite different. The marquee, with its box approach, is fairly straightforward, except for learning how to position the starting point to include the section of the drawing that you want to select—this maneuver takes practice. Once you have surrounded a section of your drawing with the marquee's dotted box, you can move that section or perform operations on it (see Figure 11).

The lasso, on the other hand, draws a line that encircles an object and then shrinks to include only the object—not the surrounding background—inside its freeform drawn rope (see Figure 12). After selecting an object, you can move it by positioning the lasso on it (the lasso will change to an arrow), holding down the mouse button, and dragging the object to the desired location. This can be particularly helpful when you are lifting one item out of a fully developed drawing or moving text without the white background.

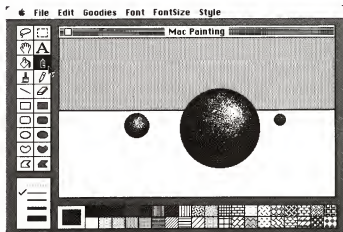
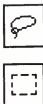


Figure 7
The spray can sprays any pattern in a realistic way, helping to create a three-dimensional effect.

The Man Behind MacPaint

When you select the *MacPaint* program, the title bar on the screen reads "MacPaint by Bill Atkinson" before you are presented with a blank canvas or a *MacPaint* document. It's unusual to see an author's name appear on the screen, and you might well wonder why this programmer deserves such an honor.

Bill Atkinson, chief programming architect of both the Macintosh user interface and the *MacPaint* program, has been with Apple since 1978. He was the first applications programmer the company hired and is now one of three Apple Fellows, the equivalent of senior vice-presidents of engineering. Bill designed the first Dow Jones stock portfolio program for the Apple II and rewrote UCSD Pascal to give the company a high-level programming language. Later he joined the original Lisa development team, where many of the Mac's features originated.

Bill describes himself this way: "I'm somewhere between an artist and an inventor. If I'd been born during the industrial revolution, I'd have used gears and pulleys. The computer is simply a more modern inventor's medium."

Since the Mac has much less memory than the Lisa, Mac programmers could not afford to waste a bit of memory space.

"Writing clean code is a science. It takes a clean, orderly mind. I was "Mr. User Interface" on the Lisa and did the QuickDraw

graphics primitives—40,000 lines of fine-tuned 68000 assembly language code. In my worst-case scenario with *MacPaint*—and believe me, I know just what it is—there are 300 free bytes left in the RAM out of the 128,000 the machine can handle. Every line of code has been made small, compact, and folded. There is no extra room."

His entire attention is focused on the machine as he works with the mouse, keyboard, and disk drives. He moves through the Mac's world with the speedy ease of utter familiarity. Bill Atkinson knows where every bit of code lives, how to get there, and what it ought to look like when it is invoked. "FatBits is the inside of the machine. I've been there and that's what it looks like. I'm still excited every time I double-click on the pencil and start scrolling along at the single-dot level."

Bill works out of his home, not surprisingly surrounded by dozens of Apple computers as well as state-of-the-art peripherals. He has a digitizer program up and running, generating dot-by-dot representations of anything that can be photographed with a video camera. "What if you could get a disk with all kinds of clip art on it? It would provide standard drawings, illustrations, and details. With the digitizer we can create such a disk. Maybe someday every computer dealer will have

Text

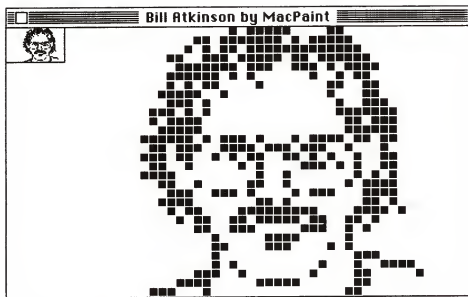
MacPaint allows you to include text in your artwork. When you click on the text icon (represented by the letter A), you have access to the wide variety of type fonts, styles, and sizes available in the Mac's resident memory (see "Chart of Typefaces"). There is certainly enough text ability to produce short captions and headings, but *MacPaint* doesn't have the sophistication of a full-scale word processing program such as *MacWrite*. Words can be written directly onto the screen at the insertion point. However, the program

doesn't provide any word-wrap feature, so unless you see that your text is going off the screen and press Return, it will wind up off the "page."

You can also write text on the NotePad or in a *MacWrite* document, and then cut and paste it into a *MacPaint* document. Any text within a *MacPaint* document can be enhanced just like any other part of a drawing. Words can be stretched, copied, moved, and manipulated in the marquee (see Figure 13).

If you type text over a pattern (or move it over a pattern with the marquee), it will be surrounded by a white border. To place letters directly on a pattern without the surrounding white space, first type some text on a blank section of the screen. Then encircle it





one to shoot particular images, as well as a whole collection of disks containing images in the public domain. You could make a copy of the original image, and then change the copied image to suit your needs on a working disk with *MacPaint*."

Before joining Apple, Bill studied neurochemistry, investigating "thought on a molecular level," in tandem with the visual arts. "I don't see code as what I do. I try to make aesthetic definitions of it. Like a sculptor using clay, a programmer has to be familiar enough with the code to shape

it, to make it do a lot of things elegantly and simply. That's the art of it."

And what does he think of the Mac?

"I like to look at the Mac in terms of the American automobile industry. Our Apple II was like Ford's Model T: the first simple, cheap, effective, but primitive mass-market vehicle. After studying the market, refining the technology, and improving the package, Ford brought out the Model A—the second great mass-market car. I think the Macintosh is Apple's Model A."

with the lasso and drag it to the desired position. (A minor aesthetic drawback is that the hollow parts of letters such as *d* and *e* are filled with white, rather than with the underlying pattern.)

Pages and Screens

The *MacPaint* screen represents about one third of a standard 8½- by 11-inch sheet of paper, which is the frame for the entire drawing. To display the whole page, you can either select the Show Page option in the Goodies menu or double-click on the hand icon. Either way, you are rewarded by a representation of the whole page, with a dotted box enclosing the portion of your drawing currently on the screen (see Figure 14). You can change the screen selection by dragging the dotted box to another part of the page or by dragging the image around the page so that a different part of it remains within the box (this method erases everything that you drag off the edge of the Show Page, however).

The Hand

Hand in hand with the Show Page option comes one of the most useful of all the screen effects: the *band*, which enables you to *scroll* the page, bringing different parts of the full drawing under scrutiny. The hand slides the page around like a transparency under an overhead projector to reveal various parts of the whole drawing. You can quickly and easily reposition the drawing without going through the longer Show Page process.

It would be useful if your drawing displayed itself immediately as you moved the hand, but you must drag the page to where you think you want to go, release the mouse button, and wait for the screen to fill in.



Keystroke Combinations

Although it takes some practice, developing facility with the *MacPaint* tools offers great rewards. When the tools are combined with various *constraint* and *shortcut* keys, a number of possibilities leap out at you from the screen. You must press these keys before you hold down the mouse button and start to move the mouse. A brief guide to these keystroke combinations can be found in the Short Cuts option in the Goodies menu (see Figure 15).

You can copy an image at various points on the screen by selecting something, then holding down the ⌘ (Command) key and dragging the copied image to another spot; the original is still there, and each time you click a new imprint is made. You can smear the im-

age, displaying one copy after another, by selecting an object and then holding down both the ⌘ and Option keys. And you can vary the spacing of the smeared copies, either by changing the speed with which you drag the pointer across the screen (faster makes them farther apart) or by selecting a different border palette option (the thinner the line, the closer the images).

An interesting set of operations comes into play when you use the Option key alone with a selection. This procedure enables you to stretch the selected image either vertically or horizontally when you drag the mouse. The result looks like a drawing on taffy, the

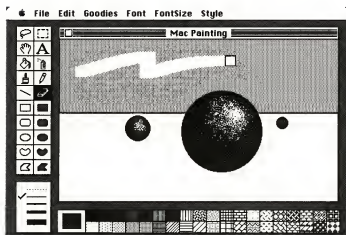


Figure 8

The eraser works like a chalkboard eraser; however, the shape of the eraser is unwieldy if you need to erase a small section of a detailed drawing.

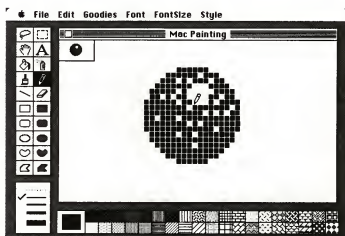


Figure 9

The FatBits feature allows you to clean up your drawings at the single-dot level.

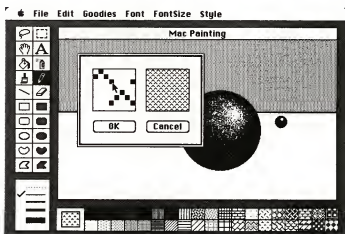


Figure 10

You can modify the dot-by-dot image of any pattern in the pattern edit window.

patterns and proportions stretched to new, and frequently bizarre, combinations. If you hold down the Shift key in addition to the Option key, you'll produce a stretched picture that stays roughly in proportion. An interesting exercise is to take a patterned image and then stretch it to watch the shifting relationships between the elements as the size changes. You can also "scroll" text through the various fonts and font sizes in a *MacPaint* document by using keystroke combinations.

Many of these keyboard controls also work with the series of geometric shapes that occupy the lower half of the tool palette. A circle, rectangle, or rounded-corner rectangle will normally be displayed in a skewed perspective as it appears on the screen. Hold-

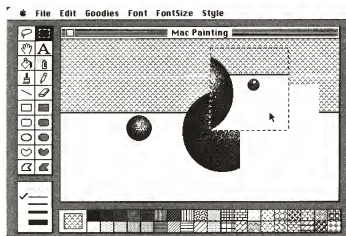


Figure 11
The selection rectangle, or marquee, allows you to select part of the drawing so that you can move, copy, or manipulate the selected area.

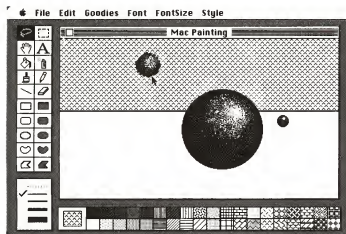


Figure 12
The lasso has a similar function to the marquee but includes only the object within the selected area, not the surrounding background.

ing down the Shift key as the image is generated will yield perfectly proportioned geometric shapes. Using the Shift key with the hand will limit motion to the vertical or horizontal, and using it with the paintbrush, spray can, pencil, and eraser will enforce the same 90-degree limits.

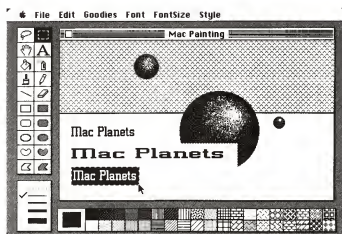


Figure 13
Text written or pasted into a MacPaint document can also be enhanced.

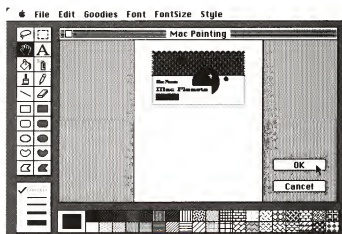


Figure 14
The Show Page option displays a representation of the entire 8½- by 11-inch page.

Graphic Design with the Mac

Bruce Charonmat

The computer age may be upon us, but magazines are still produced largely by hand. Many of the tools used in the graphic arts have not changed appreciably in decades. Lines are drawn with pens, ink, and straight edges. Shapes such as circles and squares are drawn using compasses, plastic templates, and triangles. Pieces of paper are held in place using rubber cement or wax. Mistakes are rarely forgiven; they generally require that you start over, whatever the task.

The Macintosh renders these tools and methods obsolete and will profoundly affect the graphic arts industry. While other machines equal or exceed the Mac's capabilities, they are many times as expen-

*Never before has
a computer lent itself
so well to creating the
look of a magazine.*

sive and offer neither the Mac's portability nor its ease of use. The Mac brings computer-aided design and layout skills to individuals and small publishing groups at an affordable price; it is the first truly personal graphic arts machine, and as such the Mac will flourish.

Macworld will reflect and utilize the capabilities of the Mac. Much of what you'll see in the pages of this magazine will be generated by the *MacPaint* program. This is a unique situation—never before has a computer lent itself so well to creating the look of a magazine.

The Mac can draw lines and rules in a number of widths, allowing us to draw the borders around columns. We use the *MacPaint* paintbrush in conjunction with the

Shift key to create perfectly straight lines to draw these borders. Other boxes enclosing text are created using the spray can-Shift key combination.

The large geometric shapes on the first page of each article are created with *MacPaint*, using the FatBits option for any necessary clean-up work. When all the shapes for an article are ready, we choose a *MacPaint* pattern or design a new one and use the paint bucket to fill in the shapes. The technical diagrams, flowcharts, and mechanical illustrations in *Macworld* are also created on the Mac by our crew of designers, using a variety of *MacPaint* tools. The spray can, for example, is often used to shade objects and render a three-dimensional effect, and the Grid option helps align different parts of a drawing.

Combining text with artwork is another Mac feature that we often use. Parts in mechanical drawings can be identified, instructions typed directly into flowchart boxes, and diagrams labeled in a variety of type sizes and styles. Before the Mac, this text had to be typeset separately and positioned in the illustration by hand. Now we can create the entire image at once.

We hope to show the Mac's creative side at its best in the artwork used to illustrate *Macworld*. With the advent of the Mac, the field of computer-generated art will open up to thousands of artists without previous computer experience. *Macworld* intends to serve as a forum for this expanding field, and we look forward to bringing you innovative Mac artwork each month.

▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ **Bruce Charonmat**
is the Art Director of *Macworld*.

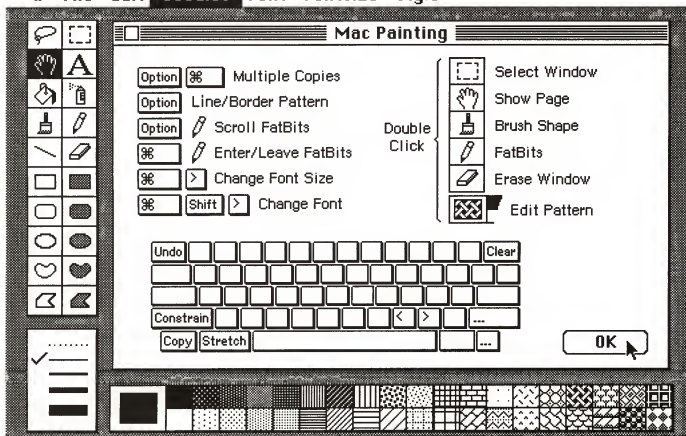


Figure 15

The Short Cuts option from the Goodies menu provides a guide to the keystroke combinations and their functions.

The Overall Picture

A *MacPaint* drawing, and any operations performed on that drawing, can have either the loose quality of a freehand sketch or the precise hard edge of an engineer's or draftsman's work. At the touch of an icon, a wide array of graphics and design capabilities become possible. It's hard to predict what effect this machine will have on architects, art directors, engineers, and artists. But for the first time an extremely sophisticated drawing program is available at a very reasonable cost, and accessible to almost anyone. Compared to the more expensive CAD/CAM (computer-aided design/manufacturing) programs available on other personal computers, *MacPaint* is much more usable, useful, and fun. It opens doors to people who could never before create art because they didn't have the "talent."

At the same time, for all its scope *MacPaint* is not a program without problems. For instance, it is very frustrating to have no full-page drawing capabilities. This means that it is impossible to draw a circle larger than one drawing screen (about one third of a page.) Furthermore, at present the program lacks the three-dimensional drawing aids essential in the design of roofs and buildings. It would also be helpful to be able to rotate a drawing or a section of a drawing (or, for

that matter, a line of text) in degrees or increments, not just the right-angle turns currently possible.

But in light of the complexity that *MacPaint* already has, all of that can't be far away. It will be interesting to see how the professional art world reacts to the program. I suspect, considering the howls of protest that always greet the new, that those with the most vested in the status quo will be the most critical. After all, if you've spent ten years setting yourself up as an expert draftsman, you're not going to enjoy the idea of a horde of schoolkids, secretaries, and stockbrokers being able to do it better than you can, and in less time. The reality, of course, is that many talented artists will flock to the Mac with its *MacPaint* program. One can hardly imagine the legacy of images that this program will produce, and I for one can hardly wait.

▲▲▲▲▲▲▲▲▲▲

Macpaint

Apple Computer, Inc.
20525 Mariami Ave.
Cupertino, CA 95014
800/538-9696; in California
800/662-9238
List price: \$195 including MacWrite

Word Processing's New Look

Andrew Fluegelman

The Macintosh represents such a major event in computing history that MacWrite, the Mac's native word processing program, must be viewed from two vantage points—one that reveals a close-up look at MacWrite's amazing capabilities and another that views the program and the Mac from a more global perspective.

Your opinion of *MacWrite* is likely to depend on whether you've ever used a computer for writing. If the Mac and *MacWrite* are your first computer writing tools, you might assume that all word processing systems let people recast sentences and paragraphs in various type styles, sizes, and fonts; change margins on the fly; see format changes instantaneously; and print exactly what appears on the screen at the click of a button.

On the other hand, if you've been using word processing programs on an Apple II, Osborne, Kaypro, IBM Personal Computer, or another computer, *MacWrite*'s abilities will probably impress you as electronic writing magic the likes of which you've never seen.

The Interface Be with You

Like all Mac applications, *MacWrite*'s foundation is the Macintosh user interface (see "The Mac Way" in "A Tour of the Mac Desktop"). The overall design and basic concepts of the program are similar to those of other Mac programs. Once you have your hand on the mouse, ready to move the pointer, you're in control of every word processing function in the program.

In *MacWrite*, the pointer takes the form of an I-beam shape within the text area. One click of the mouse button sets the insertion point at the location of the pointer, represented on the screen by a flashing vertical line. The insertion point determines where new text will appear as it is typed (see Figure 1).

Portions of text can be selected by positioning the I-beam pointer, holding down the mouse button, dragging the

pointer over a portion of text, and releasing the mouse button. Double-clicking the mouse button selects a whole word. Dragging the pointer down the left margin selects larger portions of text. As text is selected, it appears highlighted on the screen (see Figure 2). You can scroll up or down the text using the scroll bar on the right edge of the text window. The dimensions of the window can be altered using the size box, and the window itself can be dragged and moved on the screen.

Manipulating text within a document used to account for as many as half of the keystroke commands required to operate a word processing program with proficiency. *MacWrite* reduces those commands to a few simple skills using the mouse—skills that you probably learned on the Mac even before you tried using *MacWrite*. Other *MacWrite* functions are accessible through six pull-down menus listed in the menu bar at the top of the screen. After playing with the Mac for five minutes, you know how to use those commands.

As with other Mac programs, many of the features and operational concepts are intuitive; you almost know how to use *MacWrite* before you've tried it. Thanks to the standard working environment provided by the Mac's user interface, writing with a word processing program has never been so easy.

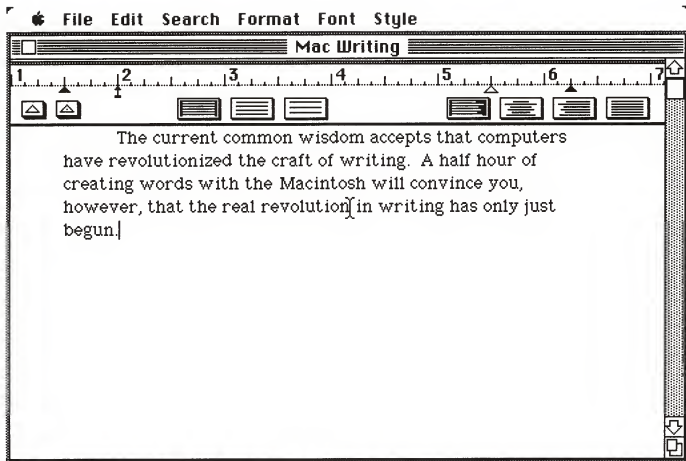


Figure 1

A MacWrite text window with an insertion point (after begun) and an I-beam pointer (after revolution)

Which Format Rules?

MacWrite is not distinguished from other word processing programs just by ease of use. It can perform some flashy screen formatting feats, all of which are controlled by the MacWrite ruler system. At the top of each document window, MacWrite displays a ruler that controls five formatting conventions: margins, indentation, tabs, line spacing, and justification (see Figure 3).

You can adjust text to fit within your chosen margins by positioning two black triangles (markers) along the ruler. In keeping with the Macintosh style of operation, you don't specify the margins with a command—you actually move the markers along the ruler by dragging them with the mouse. Whenever the margin markers

are moved, all text below the ruler is automatically readjusted to the new settings. This is a good example of the type of feature that new Mac users may take in stride, while computer veterans let out cool whistles.

A small arrow-shaped marker in the ruler governs the location of each paragraph indentation. You can also "outdent" paragraphs, which means that the body of the paragraph is indented after the first line. All indentation changes are reflected immediately on the screen.

Tabs are set in the same manner. Each new document comes with a tab marker set at the 5-inch mark on the ruler. You can reposition the marker by dragging it along the ruler; additional tabs can be set by dragging them out of the tab well (a box on the lower-left side of the ruler) and positioning them along the ruler. Decimal tabs are also provided to position typed numbers with the decimal points aligned.

Three line-spacing options are available: single, double, and line-and-a-half spacing. You select them by clicking one of three appropriate icons in the ruler bar, and the new spacing is immediately reflected on the screen. The meaning of these three icons is so obvious that you can ask new users to "choose double spacing," and they'll probably get it right on the first try.

Four other icons let you format text as left-justified, centered, right-justified, or full-justified, which makes both left and right margins even by adding letter spaces within each line. Again, you see your changes immediately reflected on the screen.

The *MacWrite* formatting features let you experiment interactively with the way your text fills the page. If you want to produce a one-page memo and find that you have three lines too many to fit on the page, just move the margin markers out a bit until the text fits. You can also reformat the memo to see what it looks like double-spaced, indented, or centered.

If you want to change formats in the middle of a document, you insert another ruler and designate new settings (see Figure 4). Ruler settings can also be copied from one part of a document to another. You can insert new rulers in the middle of a page or even in the middle of a paragraph, changing margins, spacing, and justification with a click. As you'd expect,

inserting a new ruler and changing the format settings instantly reformats the text below it.

These formatting features will have a profound effect on how you view word processing. You can control the formatting process so responsively that you won't think of your text as a static collection of characters and spaces. Suddenly, it's a fluid body of writing that can be shaped instantly to your specifications. Few writers have ever had such control over the look of their work.

The Macintosh Style

Another innovative feature in *MacWrite* is the system of specifying type styles, sizes, and fonts. At any point you can select Style from the menu bar and choose a typographic enhancement from among plain text, bold, italic, underlined, outlined, or shadow. You can mix these effects to produce bold italic underlined shadow text, or, if you're prone to hyperbole, bold italic underlined outlined shadow text. The Style menu also provides for type sizes ranging from 9- to 72-point type. Make that 18-point bold italic underlined outlined shadow text (see Figure 5).

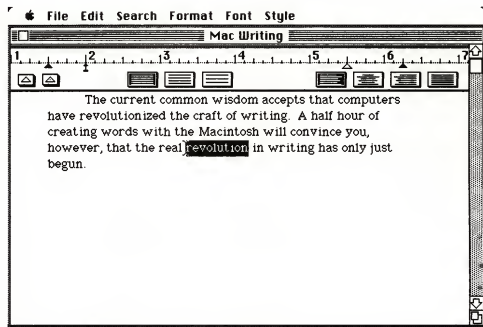


Figure 2
The word revolution selected

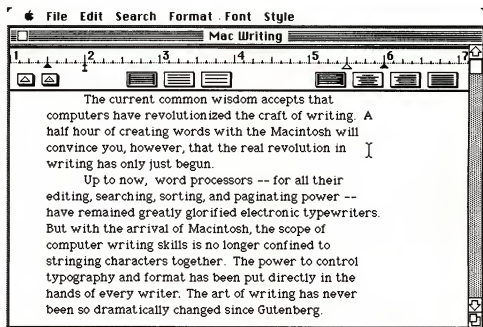


Figure 3
A MacWrite ruler showing margins at 1 1/2 and 6 inches, indentation at 2 inches, single spacing, and left justification.

Whatever styles are set define the appearance of text subsequently typed. But you can also change existing text by selecting any portion of a document, from a single character to the entire body of text, and making a selection from the Style menu. If you're already anticipating that these type changes are reflected automatically on the screen, you're becoming a convert to the Mac style of word processing.

The type possibilities may extend beyond your expectations. You can specify different type styles, fonts, and sizes within

the same line (see Figure 6). And when you later insert text in your document, the inserted text automatically appears in the specified style.

To be fair, all this praise should not be directed to *MacWrite*. These typographic acrobatics are provided courtesy of the Mac's resident Font Manager—routines that reside within the computer's permanent memory (ROM). Those font manipulations are therefore equally available to all other Mac applications. They will become

so much a part of the Mac style of computing that we will eventually come to expect them from almost any program that runs on the Mac.

The Bit of Difference

The Mac and *MacWrite* can do all these wonderful things because they treat writing as graphics rather than as just a string of characters. The letter *a* is not just the first letter of the alphabet; it's a 12-point plain New York *a*, a 10-point bold Athens *a*, a 24-point italic Venice *a*, or whatever combination of variables you choose. *MacWrite* "knows" this, and "tells" the Mac Font Manager to "tell" the screen to display a 10-point bold Athens *a*, for example (see "Chart of Type Fonts").

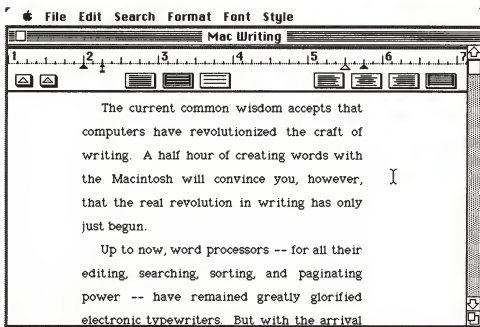


Figure 4
A MacWrite ruler with margins set at 2 and 5 3/4 inches, indentation at 2 1/4 inches, line-and-a-half spacing, and full justification.

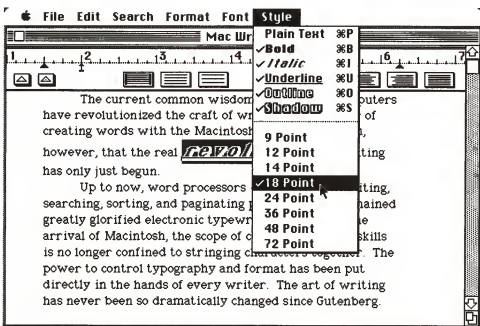


Figure 5
MacWrite's Style menu, with the word revolution set in 18-point bold italic underlined outlined shadow text

The screen can display the letter in this form because the computer has control of each of the 175,104 dots that make up the total image on the screen. The Mac screen is bit-mapped, meaning each of those dots is controlled in the Mac's memory. Thanks to the Mac's ability to display text in bits, every *MacWrite* writer has been given the power to be his or her own typographer and graphic designer.

The Limit

Now for a dose of reality. Based on the introductory state of the Mac, each *MacWrite* document is limited to about 27,000 characters, or about 5000 words, which is roughly the length of this review. It's about 20 double-spaced pages in 12-point type. That's certainly more than enough space to write your next office report or a long, chatty letter, but you might find yourself

running out of space just as you're keying in the climactic ending of your short story.

The immediate reason for this document size limit is that a good chunk of the Mac's random-access memory (RAM) is taken up by the programs that control the Mac's desktop and the *MacWrite* program itself. The remaining RAM is what you have available for your document.

The real limitation, however, is the fact that *MacWrite* holds the entire document in memory while it is being worked on. There are other methods by which word processing programs handle documents, such as by writing portions of the document back to disk when necessary. This method permits a document length limited only by disk storage space. In computer parlance, such a storage system is said to be *disk-resident*, as opposed to the *memory-resident* system that *MacWrite* uses.

If you've got a lengthy writing project and can't manage to chop your material into 5000-word chapters, two solutions to this memory-bound limitation are possible.

One is to wait until more memory becomes available for the Mac. If a Mac had 512,000 bytes of memory installed, the maximum *MacWrite* document would theoretically be about 400,000 characters, which is the size of a short novel. But that might be too long a wait for your immediate writing needs.

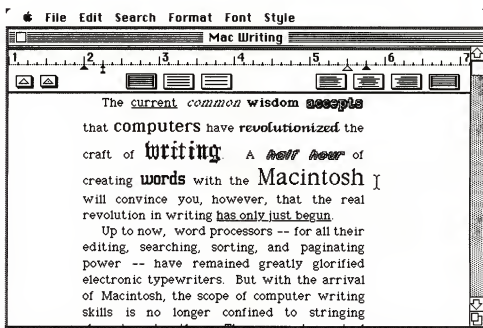
The second solution is to wait until a word processing program is introduced for the Mac that uses a disk-resident document system. Several such programs are under development and forthcoming. By the time you read this, you might not have to wait at all.

Old Traditions

Now that you've been wowed by *MacWrite*'s features and have come to terms with its Achilles' heel, how well does it perform the standard operations that most word processing programs provide? The following is a brief rundown of some of the essentials.

Insert and overwrite. All text is inserted in the document as it is typed at the insertion point. You cannot overwrite characters—that is, type characters in place of existing ones. Limiting *MacWrite* in this

Figure 6
A variety of *MacWrite* type styles and fonts in a single document



way is a wise choice, given the general Macintosh edict that application programs should not place users in different modes of operation.

Backspace. The Backspace key moves the insertion point one character to the left while deleting the character to the left; in other words, it is a *destructive* backspace. Seasoned writers and poor typists probably use this key more than any other.

Delete. The *MacWrite* program provides three ways to delete text. The Backspace key deletes destructively as described above. You can also delete by selecting a portion of text and choosing Cut from the Edit menu. The cut text then resides in the Clipboard, ready for further use.

The third deletion technique is the most innovative and also the most useful. If you select a portion of text and start typing new text, the selected text will be deleted and the new text inserted as you type, all in one automatic operation. It's the handiest way to make a spot correction. Pressing the Backspace key when text has been selected deletes that text without cutting it to the Clipboard.

Move and copy. Moving a block of text in *MacWrite* is accomplished by selecting the text, cutting it to the Clipboard via the Edit menu, choosing a new insertion point, and then pasting the text from the

Clipboard into the document (see Figure 7). As an alternative, text can be copied to the Clipboard, leaving the original text in place and making the Clipboard text available for pasting elsewhere in the document.

This moving operation is fairly efficient in that it can be accomplished entirely with the mouse, but it does require that you move from the text area to the menu bar and back again twice. An alternative method that uses the ⌘ (Command) key is a very efficient way to make these moves (see "Keyboard Control").

Search and replace. The Search option on the *MacWrite* menu bar has commands for finding a specified string of characters and either selectively or globally changing the "find" string to another string (see Figure 8). Both functions ignore case; that is, if you are searching for *Mac*, you will find *Mac*, *mac*, *Macys*, and *supremacy*.

MacWrite's search and replace function is somewhat limited. Other word processing programs allow you to specify whether you want to ignore case. A further refinement is whether the replace string is inserted in a "case-sensitive" manner. In other words, if you are finding *Mac* and

changing it to *Sack*, both *Mac* and *mac* will be changed to *Sack* and *sack* respectively. A flexible range of search and replace options is the sort of detail that in the traditional word processing world distinguishes a good program from a great one.

Bells and Whistles

The word processing world is filled with other fancy features. Some programs let you store different formats for memos, letters, and outlines and implement them with a single command; edit several documents at once using multiple windows; create footnotes, tables of contents, and indexes automatically; or define certain keystrokes to type whole words, phrases, or sentences or to define a series of program commands into "macro" commands. Some programs even check your grammar and spelling. *MacWrite* does not offer any of these extra treats.

MacWrite does offer one rare and welcome feature, however. The Undo command in the Edit menu always gives you the option of undoing your last action. Thus, you can undo deletions, moves, style changes, and even the last text typed as an insertion. It's like having a word processing seatbelt, only more effective, because you're always wearing it and it can take you back in time, before the accident occurred.

What's more, in classic Macintosh style, the text of the Undo command changes to reflect your last action. For example, if you are undoing a Cut command, the prompt will say Undo Cut. If you choose Undo twice in succession, the second time the prompt will say Redo Cut and undo your previous undo. It's a powerful dose of Mac magic.

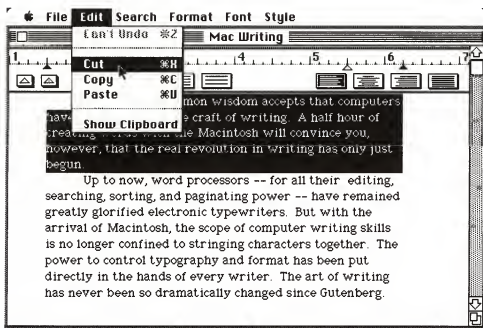



Figure 7
Choosing the Cut command from the Edit menu to cut selected text into the Clipboard

Up to now, word processors -- for all their editing, searching, sorting, and paginating power -- have remained greatly glorified electronic typewriters. But with the arrival of Macintosh, the scope of computer writing skills is no longer confined to stringing characters together. The power to control typography and format has been put directly in the hands of every writer. The art of writing has never been so dramatically changed since Gutenberg.

Heads and Tails

Like many traditional word processing programs, *MacWrite* provides the ability to specify headers and footers—text that appears at the top or bottom of each page of the document. In some respects, these features are powerfully implemented. You can include several lines of text in both the header and footer, and you can set the type style, size, and even the font for the headers and footers independently of the main text. The headers and footers are specified in separate windows, but their effect is displayed as part of the text on the screen (see Figures 9 and 10).

There are some problems with this feature, however. The header and footer windows provide three icons that can be dragged into the text for setting the page number, date, and time. Unfortunately, these icons do not display their effects in the header window—you have to switch to the text window to see the page, date, or time; thus, aligning and editing them correctly can be difficult (see “Tips for Mac Writers” for some techniques to alleviate this problem).



Every MacWrite writer has been given the power to be his or her own typographer and graphic designer.

The headers and footers also control the top and bottom margins of each page. If you want to create a larger top margin but don't want any header text, you have to open the header window and insert one or more Returns. This technique produces no discernible effect on the header screen, so you have to guess what effect you're producing. *MacWrite* will not let you insert more than seven lines in a header or footer. If you want more than seven lines' worth of top margin, you have to select line-and-a-half or double spacing on the separate header/footer ruler before you enter the invisible Returns or specify a differ-


Keyboard Control


A debate has raged for years in word processing circles concerning proper use of the keyboard. One school, consisting mostly of writers who struggle with the typewriter keyboard using four fingers at best, has cried out for special function keys that could be dedicated to specific, convenient word processing commands.





The other school, which seems to be made up entirely of touch-typists who generally type faster than the national speed limit, wants to keep ten fingers on the keyboard as much as possible.


The Mac has no function keys, and it's probably a good thing. Once you deal with more than 107 of them and their various permutations, remembering which convenience each key provides becomes a chore. All the Mac's commands are accessible via the mouse pointer and “soft” checklists, buttons, and boxes that appear on the screen when requested or when the situation warrants. It's a much more sane and practical system.

Nevertheless, there will surely be some proficient typists among the ranks of Mac users, and they will probably prefer to remain keyboard bound while they write. To accommodate them, *MacWrite* provides the ability to invoke

some commands directly from the keyboard, using the Command key  in conjunction with an alphabetic key. These keyboard alternatives are conveniently listed next to the equivalent commands in the menus.

Two systems of keyboard alternatives are provided within the *MacWrite* program. One permits you to select type styles by holding down the  key and pressing p, b, i, u, o, or s for plain, bold, italic, underlined, outlined, and shadow text, respectively. Using this method is very practical; for instance, you can italicize a word and return to plain text without having to break your keyboard flow.

Other keyboard alternatives permit you to Undo the last action by pressing -z. Cut selected text to the Clipboard using -x, Copy using -c, and Paste with -v.

If the mnemonics of these letters escape you, you'll notice that the z, x, c, and v keys occupy the bottom row of the keyboard. Thus, you can easily invoke these four -key options with one hand. It's a technique that Mac writers should quickly develop, because it cuts down on mouse activity and makes text manipulation more efficient.

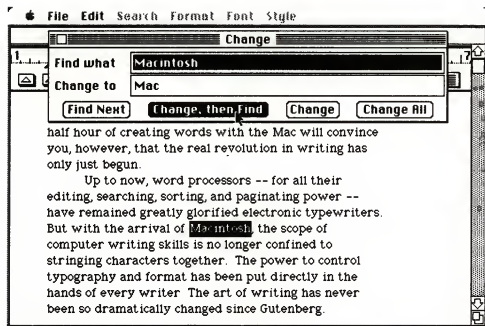


Figure 8
The MacWrite Search/Change menu

ent type size for the header. This is a level of abstraction that few users will enjoy mastering.

Many traditional word processing programs provide more header and footer options, such as positioning the header or footer on the right or left depending on whether the page number is odd or even. This feature is handy for reports duplicated by printing on both sides of the page. Most programs permit you to specify different header text midway in a document. *MacWrite* does not, although it does let you specify a page as a "title page," which maintains the header spacing but does not print the header text.

Blind Formatting

Some of the same difficulties encountered in formatting headers occur when you format the main text. All of the formatting effects are based on the position of Returns in the text and the points at which type styles change. So long as you keep the text in orderly paragraphs and make rational typographic choices, *MacWrite* will present a fairly clear picture, and what you see is in fact what you get.

You can confuse your document, however, by mixing spaces with Returns and type style changes; figuring out where you have to insert or delete to make your text bold instead of underlined or indented instead of outdented can become very mysterious indeed. A welcome option would be a menu choice that temporarily displays where the formatting breaks occurred.

Another formatting reservation concerns the ruler-like markers. While they do make page formatting simpler than in almost any other word processing program available, they can get in the way and become ungainly if your document has several format changes. (Rulers can be optionally hidden.) Centering a line of text in *MacWrite* is also rather cumbersome (you have to insert a new ruler just for that purpose).

Another format disappointment is that the maximum width of a *MacWrite* document is about $\frac{6}{8}$ inches of text area—what can be displayed in the *MacWrite* window. It's not unusual for word processing programs to be able to scroll horizontally to accommodate wide documents. There is a further limitation: if you have reduced the dimensions of the *MacWrite* text window, you cannot scroll horizontally to see even the $\frac{6}{8}$ -inch width. This limitation is a major departure from the Macintosh user interface.

Reading and Writing

Many word processing applications involve mixing and merging different documents, often portions of standard text called *boilerplate*. You can accomplish this task in several ways with *MacWrite*, but the program lacks the two features necessary to do the job directly: the ability to read another saved file into your document and the ability to select a portion of text and save it as a separate file on disk.

The easiest way to transfer portions of text from one document to another with *MacWrite* is to Cut or Copy a selection to the Clipboard, save the existing file, open a new document, and then Paste from the Clipboard into the new document. This process works, but would be very inconvenient if you needed to insert several documents into a text file or pull several selections from text and save them as separate documents. The File menu offers the options of getting a copy of a document or duplicating the current document by saving it with another name, but both of those operations work only on whole documents.

The best way to do multiple document reads and writes with *MacWrite* is to make use of the Scrapbook, a feature supplied as a desk accessory. You can place several selections into the Scrapbook and then insert them from the Scrapbook into new documents. This procedure is somewhat more efficient, but still requires two steps to complete one operation (see Figure 11).

The Graphics Connection

MacWrite doesn't quite measure up to some of the traditional programs in the file transfer department, but how many traditional word processing programs let you combine graphics with text? With *MacWrite*, *MacPaint*, and a little practice, you can not only insert a drawing into your document, you can change its size and proportions while it's on the page. Word processors just aren't supposed to be able to have that capability.

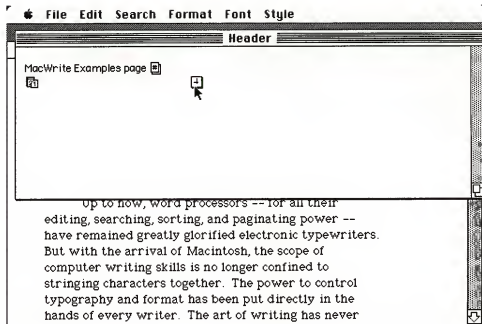


Figure 9
A header window
showing the page
number, date, and
time icons

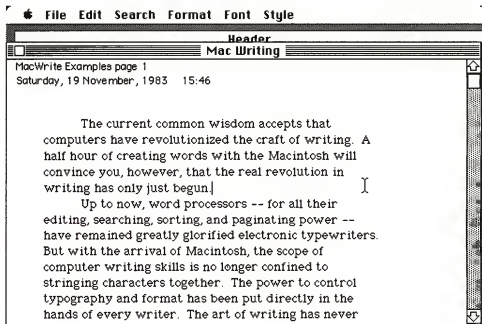


Figure 10
The effect of the
header window settings specified in
Figure 9

Printing

Printing a word processing document has long been the writer's most dreaded chore. The problem was that you could never be really sure how the document would print while you were producing it on the screen; you had to connect a multitude of printing commands and specifications with the document for it to print correctly.

Enter *MacWrite*. Choose Print from the File menu, answer some questions about page length and printing quality, and watch your printer produce exactly what

appeared on the screen (see Figure 12).

The only exception to this process is that if you choose high resolution from the Print menu, your printed document will look even better than it did on the screen.

The printing features have some peculiarities. The Print menu offers the options of printing standard 8½- by 11-inch letter size, legal size, and European size documents in a tall or wide format, but as noted above, the *MacWrite* screen limits the text portion of a document to a width of 6¼ inches.

The printing function also lacks one feature that would have gone a long way to ameliorate the document size limitation. The ability to chain print separate documents into one long printed document is not an uncommon word processing feature. If this feature were included as a *MacWrite* option, you could produce final printed pieces of virtually any length with minimal inconvenience.

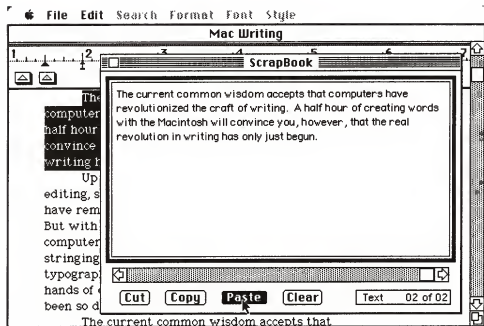


Figure 11
Pasting text to the
Scrapbook

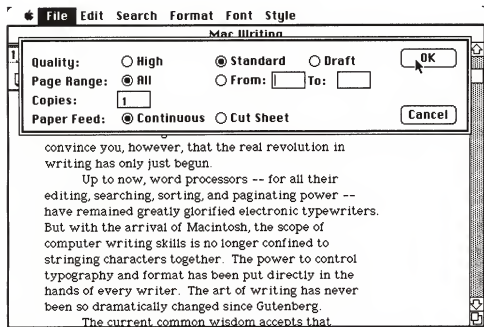


Figure 12
The File/Print menu

Two Reviews

Now it may be more apparent why this review is really two reviews. Compared to the mundane functions of the word processing programs that preceded it, *MacWrite* holds its own as a perfectly serviceable, if somewhat limited, word processing package. It's an appropriate program for beginners because of its simplicity. Some features could have been better designed, and many bells and whistles exist in the word processing world that *MacWrite* doesn't pretend to offer.

But for new computer writers and word processing veterans alike, *MacWrite* also provides the first widely accessible version of a revolutionary new style of computer writing tool. It excels in the areas of screen formatting and type selection. *MacWrite* does amazing things effortlessly that most other programs can't even attempt. Without question, no word processing program has ever offered so much power with so much simplicity to such a wide audience.

This is why every writer, regardless of his or her computer experience, should be thrilled by *MacWrite*. The Macintosh is

going to change the way we think about written communication. *MacWrite* is the first popular program to show us why and how.



MacWrite
Apple Computer, Inc.
20525 Mariani Ave.
Cupertino, CA 95014
800/538-9696; in California
800/662-9238
List price: \$195 including *MacPaint*

Multiplan Meets the Mac

Andrew T. Williams

The Mac teams up with an improved version of Microsoft's venerable electronic spreadsheet program.

Last year Microsoft's *Multiplan* was named "Software of the Year" by *InfoWorld*, a widely read and respected news weekly for microcomputer users. The honor was well deserved because *Multiplan* is a powerful piece of software. This year *Multiplan* has teamed up with the new kid on the block, the flashy Macintosh. The result is definitely something to behold. Later in this review, you'll see exactly what this marriage of technical innovation and tried and true software means in terms of specific features. But for those users who are new to computers, here's a brief introduction to the world of electronic spreadsheets.


Electronic Spreadsheets

To understand an electronic spreadsheet, you should begin with its more familiar ancestor, the manual spreadsheet. Manual spreadsheets (sometimes called worksheets or accountant's pads) are common tools used in business analysis. They can be large, complicated affairs containing hundreds of interrelated entries, or they can be as simple and straightforward as a cash register receipt from the local supermarket.

Accountants use spreadsheets to keep the books, managers use them to prepare budgets and sales projections, and financial analysts use them to organize their analyses. Everyone uses them to keep records.

An electronic spreadsheet adds the power of a microcomputer to the framework of the manual spreadsheet. It gives each spreadsheet the natural calculating ability of the computer so that formulas and functions can be entered into the cells of the worksheet, instead of just the numbers and text available on a manual spreadsheet.

Using an electronic spreadsheet, you can construct tables that accurately model the relationships between elements on the spreadsheet and use the computer's rapid calculating powers to recalculate the worksheet whenever new information is entered or existing information is changed. To be sure, information is still arranged in the cells of the familiar spreadsheet grid



Multiplan's linking feature allows you to combine several worksheets into a gigantic workspace.

formed by the intersection of columns and rows, but the size and power of the spreadsheet is greatly expanded.

A 13-column by 30-row manual spreadsheet has 390 cells, enough for 30 line items for twelve months plus a column for the row headings. *Multiplan*, by contrast, has 16,065 cells arranged in 63 columns and 256 rows. With *Multiplan* you can do five-year projections by months or fifteen-year projections by quarters. *Multiplan's* linking feature allows you to combine several worksheets into a gigantic workspace.

20000/CPI

Options Calculator

2

CPI

3

PURCHASING POWER OF THE DOLLAR

CONSUMER
PRICE
INDEXHOW MUCH YOU'D HAVE TO
EARN TO EARN \$20,000
IN 1967 DOLLARS

1.247



\$16,038.49

1.127

\$17,746.23

1.058

\$18,900.00

0.86

\$23,255.00

0.621

\$32,206.10

0.406

\$49,261.08

0.348

\$57,471.26

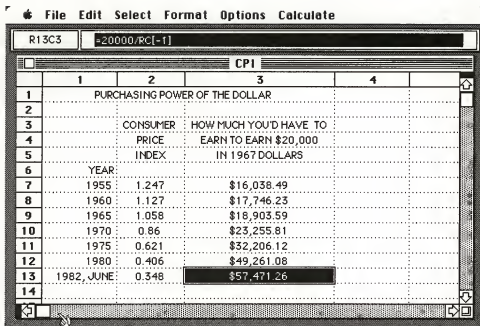


Figure 1
Sample Multiplan
spreadsheet docu-
menting the sad
decline of the pur-
chasing power of
the U.S. dollar.

More important than size alone is the fact that *Multiplan* resides in the computer's memory. The Mac is very good at keeping track of things, doing error-free calculations, and letting you change the appearance of the display.

The speed of a *Multiplan* spreadsheet calculation clearly manifests the power of electronic spreadsheets. A simple example is adding up a column of numbers. *Multiplan* has a function that does this in less time than it takes to dot an *i*. You can change a number in the list, and the program will recalculate the total (along with every other formula and function on the worksheet) and display the new answers in a flash.

"What If" analysis—changing a few numbers to see what will happen to a budget, a balance sheet, or any other table—is an important financial analysis tool. It is one of the tasks that *Multiplan* and other electronic spreadsheet programs do best. The results of a few simple changes that may take hours of manual calculation (always subject to human error) are presented error-free by the computer in seconds.

Another important feature of an electronic spreadsheet is its ability to insert or delete space in the middle of an already

constructed table. Never again will you have to use scissors and paste to remove an unwanted row or column. Never again will you need to resort to microscopic writing to include a line item you missed or an item someone suggested you consider. Furthermore, every formula and function on the worksheet adjusts automatically to accommodate the new rows or columns.

Electronic spreadsheets, along with word processing programs, are what keep microcomputers from being merely expensive toys. These programs give you great control over two of humankind's most essential means of communication and analysis: words and numbers.

Also, spreadsheets and word processing programs do not require that you use specific tools to solve problems in pre-determined ways. Instead, they provide two general types of electronic "blackboards" and "chalk boxes." These are generalized tools that let you decide how best to solve your particular problems based on your individual knowledge and creativity.

Macplan

Multiplan is the first spreadsheet program available for use on the Mac. But it is such a different version of the well-known product that those familiar with it might be tempted to call it "MacPlan." If you have used *Multiplan* before, you'll recognize the program, but there are many new—and worthwhile—things to learn. The uniform working environment of the Mac—windows, icons, pull-down menus, and the like—and the use of the mouse for cursor control have important, positive effects on the way *Multiplan* operates.

Figure 1 shows a typical *Multiplan* window. It displays a table documenting the sad decline of the purchasing power of the U.S. dollar. The menu bar at the top of the window lists the *Multiplan* menu titles: Folder, Edit, Select, Format, Options, and Calculate. Most of the program features are selected by choosing items from one of the menus. Each menu can be pulled down by locating the pointer on the desired menu title and pressing and holding down the mouse button.

You can select items from the menu by moving the pointer to the appropriate item and releasing the mouse button. The

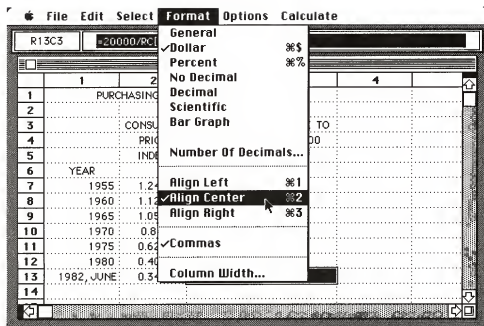


Figure 2
Sample Multiplan menu. The checks opposite Dollar, Align Center, and Commas indicate that those three formats control the display of the number in the active cell.

checks opposite Dollar, Align Center, and Commas in Figure 2 indicate that those three formats control the display of the number in the cell shown in black in the body of the worksheet.

The line below the menu bar is the *formula bar*. The left side of the formula bar contains the address of the active, or current, cell in the body of the worksheet. In Figure 1, the active cell is at the intersection of row 13 and column 3. Hence, the address of the active cell is R13C3. Each cell on a *Multiplan* worksheet has a unique address made up of the row number and column number of the cell's location.

The active cell is the only cell in which information can be placed or modified. It is displayed as black with a solid white border, making it easy to spot. The contents of the active cell are displayed on the right side of the formula bar. In Figure 1, the active cell contains a formula ($=20000/RC[-1]$), even though it displays the number \$57,471.26 in the body of the worksheet. The formula tells *Multiplan*

what it must do to find out how much you would have needed to earn in June of 1982 to have the average purchasing power of 20,000 1967 dollars.

The denominator of the expression, $RC[-1]$, is how *Multiplan* refers to cells within formulas. It means, "Get the value from the cell in this row (R), but one column to the left ($C[-1]$)."

This type of cell reference can become very complicated when you are referring to ranges, and it is a major reason *Multiplan*'s ability to assign names to ranges is so essential.

The Mouse

One of the prominent features of the Mac is the mouse. While you'll need some practice to get the right hand-eye coordination to move the pointer with the mouse, the power and convenience it brings to *Multiplan* are well worth the effort. (Some commands can be implemented from the keyboard. These key combinations, which are listed next to the equivalent commands in the menus, are convenient when you are keying in data.)

The action controlled by the mouse depends on the location of the pointer on the screen. To help you distinguish the actions, the pointer changes shape as the ac-

tion changes. When the pointer is in the menu bar, it is arrow shaped and can be used to pull down menus. When the pointer is in the body of the worksheet, it is a cross and is used to activate cells. In the formula bar the pointer is an I-beam and indicates where editing will take place.

If you move the pointer to one of the vertical lines that separate the column numbers, the symbol changes to indicate that you can change the column width by dragging the boundary to the right or left. (Column 3 in Figure 1 has been expanded to accommodate the large column heading.) When an action requires a few moments to complete, the pointer changes to a wristwatch to indicate that you must wait.

While they do take some getting used to, the context-sensitive symbols are an excellent visual guide to what will happen when you work in different areas of the *Multiplan* window. The mouse is a convenient way to choose options from a menu, and can quickly accomplish such routine tasks as selecting a range of cells, changing column widths, or splitting the screen into separate window panes.

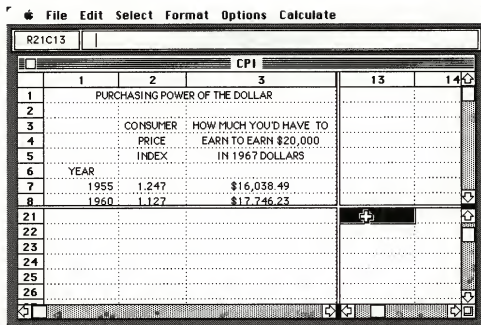


Figure 3
Multiple windows
allow you to view
different sets of cells.

Scrolling the Worksheet

Only a small fraction of the *Multiplan* worksheet appears in the grid in the lower part of the screen in Figure 1. Although the number of cells displayed depends on the width of each column, usually fewer than 100 cells are visible on the screen at any time. Actually you have plenty of room to work with, since the worksheet in the Mac's memory has 16,065 cells.

The general procedure for making different cells appear on the screen is *scrolling* the worksheet. This term refers to the way the worksheet appears to wind and unwind on the screen. Since you will use scrolling frequently when you are constructing or using an electronic worksheet, the way a program handles this function is very important. One of the Mac's greatest strengths is that it performs these operations efficiently.

The easiest way to expose cells that are just off the screen is to position the pointer at the edge of the grid. Pressing and holding the mouse button while you move the pointer into the border causes new rows or columns to appear.

If you want to view distant cells, the Mac has some particularly efficient ways to get you there. *Multiplan* uses the scroll bars, scroll boxes, and scroll arrows that are standard equipment on most Mac windows (see "A Tour of the Mac Desktop" for a detailed description of these features). By

using the pointer to position the scroll boxes in the horizontal and vertical scroll bars, you can quickly and easily cause any of *Multiplan*'s 16,065 cells to appear on the screen.

Window Bars

Another feature of *Multiplan* is the *window bar*. This is the small black rectangle at the top of the vertical scroll bar and at the far left of the horizontal scroll bar. You can position the pointer in a window bar, hold down the mouse button, and drag the window bar along the scroll bar. The screen will split into two windows at the point where you release the mouse button (see Figure 3).

Unlike a similar feature on other electronic spreadsheets, you cannot scroll all windows independently. Windows above one another must display the same columns, and windows beside one another must display the same rows. However, windows on the diagonal can be scrolled independently. You can view different sets of cells in the upper-left and lower-right windows. Thus, you can display an entry space (a place where you enter parameters) in the upper-left window and, at the same time in the lower-right window, a set of far-removed cells that form display output.

Selecting Cell Ranges

Many of *Multiplan*'s built-in functions and commands operate on all cells in a group or range of cells. The SUM function is an obvious example. It adds up the numbers in a range and displays the answer in the cell containing the function. Another example is the Format command, in which all the cells in a range can be assigned the Dollar format or aligned to the center of the cell, for example.

To operate on a range of cells, the program must provide ways to select the cells. *Multiplan* gives you several ways to accomplish this task. In fact, there is nothing like it on any other electronic spreadsheet program. You can select an entire column or row by positioning the pointer on a row or column number and clicking the mouse button. Like all selected ranges, the selected row or column will appear in reverse video (white characters on a black background).

Or you can position the pointer on the first cell of the range, hold down the button, and drag the pointer to the opposite corner of the desired area. The area may

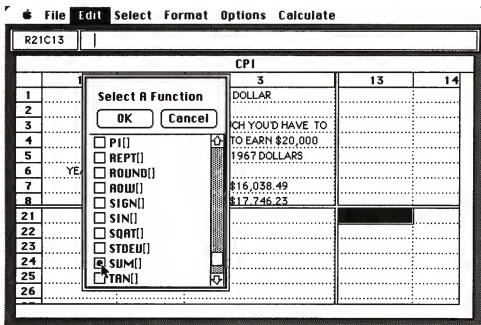


Figure 4
The Paste Function command. You use the scroll bar in the dialog box to view the entire list of Multiplan's built-in functions.

be a block of cells as well as a row or column of cells. If you drag the pointer into one of the borders, the screen will scroll and the selected area will expand until you release the mouse button.

You can also select a block of cells by using the Shift key in combination with the mouse button. First, designate the cell at one corner of the block by clicking the cell. That cell is the active cell, and it appears in reverse video surrounded by a solid white border. Next, move the pointer to the cell at the opposite corner of the block. Then press the Shift key and click the mouse button. The entire block will be selected, and the cells will be displayed in reverse video.

This method of selecting a range of cells works especially well when the opposite corner of the block is off the screen. Just use the scroll bars to display the opposite corner on the screen and designate the block by using the Shift key and the mouse button. If you make a mistake, reposition the pointer and Shift-click again. The original cell will remain active to define the opposite corner of the block.

Another method for selecting a group of cells uses the ⌘ key in combination with the mouse button to randomly select

nonadjacent cells. Each time you click the mouse button when the ⌘ key is pressed, a cell is selected. If you hold down the mouse button, you can drag the pointer over a group of cells. If you hold down the ⌘ key and the Shift key at the same time, you can select a block of cells.

Any of these groups of cells can be assigned a name and used in a built-in function or a *Multiplan* command. Furthermore, you can move through any selected range by pressing the Enter key, or back up through a range by using the Shift-Enter key combination. The combination of the Enter key and *Multiplan*'s cell selection options gives you a powerful worksheet tool.

A word of caution. The Shift key and the ⌘ key are disabled whenever the Caps Lock key is engaged. If *Multiplan* isn't letting you use all of the selection features, check to make sure the Caps Lock key is not engaged.

Special Editing Commands

Multiplan's Edit menu has several special capabilities not found in other spreadsheet programs. One particularly useful feature is Undo. If you select the Undo command, what you have just entered will be undone. You can also click the "stop sign" that appears in the formula bar when you begin an entry. The stop sign cancels a command, formula, or other entry in mid-stride.

Two other convenient Edit commands are Paste Function and Paste Name. Figure 4 shows what happens if you select the Paste Function command from the Edit menu. You are presented with a dialog box listing all of *Multiplan*'s built-in functions. You can use the scroll bar in the dialog box to view the entire list. When you find the function you want, you can select it by

You can protect either a range of cells or the entire worksheet.

clicking the box in front of the formula. The selected function appears in the formula bar, ready for you to designate the range of values on which it will operate.

The Paste Name command operates in the same way. When you select it, you are presented with a dialog box containing all of the names you have created. You select the one you need, and it is pasted into the appropriate place.

The Fill Down command fills a range with the contents of the first row, while the Fill Right command fills a range with the

contents of the first column. To begin, you must specify the range to be filled. A *Multiplan* range is easily seen, because every range is displayed in black as it is selected.

The Fill commands perform the same function as the Replicate or Copy commands found in traditional spreadsheet programs. These commands are constantly used to spread functions, labels, or data from one cell or set of cells to the rest of the worksheet. Using these commands makes constructing a spreadsheet easy and fast. In most situations, using *Multiplan's* Fill commands is much easier than using the Replicate or Copy commands of other spreadsheet programs.

Protecting Worksheets

Multiplan can protect worksheets from accidental changes. This feature can save your valuable worksheets from your clumsiness and careless mistakes. It can also prevent your worksheet from being damaged when your well-meaning assistant decides to update it.

You can protect either a range of cells (which may or may not be adjacent) or the entire worksheet. If you protect the entire worksheet, use the Unprotect option to open up the cells into which data will be entered.

Multiplan has several features that make the protection option especially useful. First, protected worksheets are displayed on the screen without the row and column numbers and the dashed lines that mark the cell boundaries. You immediately know that you are working with a protected worksheet. Secondly, *Multiplan* designates any unprotected cells by underlining them so you can see exactly which cells are available for data entry.

Multiplan goes the standard protection features one better by having you assign a password to each protected worksheet. Once a worksheet is protected, only using the password can unprotect it, so be sure to pick a password you'll remember.

Finally, you can move from one unprotected cell to the next on a protected worksheet by pressing the Enter key. This procedure makes it easy to update a protected worksheet. Simply type the data for the active cell and press Enter. The active cell then moves automatically to the next unprotected cell, ready for your next data entry.

Linking and Naming

The Macintosh version of *Multiplan* carries over two features for which the program has become famous: the ability to link several worksheets and the use of English words, instead of row and column cell references, as names for ranges.

The linking feature, from which *Multiplan* gets its name, allows you to establish a connection between supporting and dependent worksheets. Linking is the natural way to combine quarterly reports into annual reports or tables of regional or divisional sales figures.

The link between worksheets is accomplished by assigning a name to a cell or range of cells you wish to link from one worksheet to another. After you specify the ranges, use the Copy command from the Edit menu to establish the link. Once the link has been established, the dependent worksheet will copy the information from the supporting worksheet each time the dependent worksheet is loaded into the Mac's memory. If you want, you can break a link with the Undo Copy command in the Edit menu.

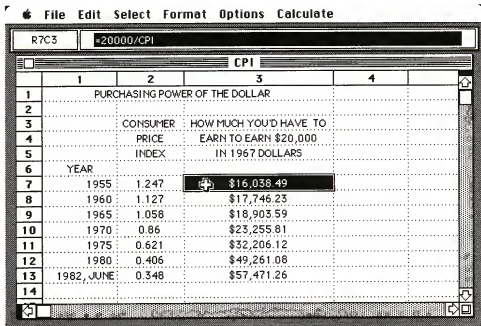


Figure 5
Figure 1 with the formulas displayed. In column 3, the name CPI is used in place of cell references in the formulas.

In addition to their use in linking worksheets, named cells or ranges of cells are convenient to use with all spreadsheet functions. In Figure 5, the Consumer Price Index in column 2 of the spreadsheet in Figure 1 has been given the name "CPI." This name is used in place of cell references in the formulas in column 3. The formulas are much easier to read and understand when CPI replaces the cell reference, RC[-1]. This procedure is even more useful when a name replaces a range of cells in a formula or built-in function.

You can use the Name option from the Select menu to view all of the names you have assigned and to see which cells are assigned to each name. The Paste Name command in the Edit menu allows you to paste a name into a formula without having to re-type the name.


Entering Data and Text

The Mac version of *Multiplan* has a good method for entering information into worksheet cells. First, you use the pointer to define the range into which you wish to make your entries. After selecting the range, you type your entry and press Enter. The active cell, which is outlined in white, will automatically move to the next cell in the range.

You can skip cells by pressing Enter, and you can return to cells in which you have already entered information by simultaneously pressing the Shift and Enter keys. This technique works with the Paste Function and Paste Name commands, as well as with numbers, formulas, and text. You must be careful not to use the Return key instead of the Enter key; pressing Return will cancel the range.

Flaws in the Plan

Set off against an impressive list of strengths are three problems. The first is *Multiplan's* inability to scroll each pane of a split screen independently. As noted, only diagonal windows can be scrolled independently, and independent scrolling is necessary to view widely separated sections of a worksheet. The split screen is



The relatively small size of the Mac screen is a drawback.

most often used when you are performing What If analysis with large tables, and non-independent scrolling means wasting about half of an already small screen to obtain the two independent diagonal windows.

The relatively small size of the Mac screen is also a drawback. When *Multiplan* first appears on the Mac screen, it displays 84 cells in 14 rows and 6 columns. In contrast, *Multiplan* displays almost 75 percent more cells—140 in 20 rows and 7 columns—on the IBM Personal Computer. Having more cells on the screen at one time makes working with a worksheet easier. In fact, the major selling point of some microcomputers is that they display 132 characters across the screen at one time and are therefore well suited for working with spreadsheet programs.

The final problem is related to the Mac's uniform working environment, not to a specific application program. When you want to erase files, including *Multiplan* worksheet files, the Mac requires that you take them from the desktop and place

them in the Trash. This operation must take place at the disk window level either before you enter *Multiplan* or after you have left the program.

The use of icons and a uniform way of doing similar tasks in all application programs is certainly one of the Mac's greatest strengths; it helps you move from one application to another easily and with a minimum of training. But in this case, the uniform environment is a drawback. Most people want to have one last look at a worksheet before consigning it to electronic purgatory, and with the Mac you can't do that. What you must do is keep a list of the files you want to discard in the Trash, and then do the operation after you've left *Multiplan*. Microsoft should definitely put a Delete File command in the File menu.

These few objections aside, the most exciting thing about *Multiplan* on the Mac is the way the classic strengths of a solid spreadsheet program team up with the latest technology to produce a winning combination. Many people will undoubtedly find using the mouse an easy and natural way to select cells, split and scroll windows, and edit formulas. The pull-down menus and dialog boxes are more important features, however, because they make *Multiplan* easier to use than ever before. And you simply cannot make a spreadsheet program too easy to use.

▲▲▲▲▲▲▲▲
Multiplan
Microsoft Corporation
10700 Northup Way
Bellevue, WA 98004
206/828-8080
List price: \$195



Apple's Imagewriter Printer

The Imagewriter mirrors the Mac screen's text and graphics

Lon Poole

What you see is what you get—that's the watchword of the Apple Imagewriter printer. Like the Mac screen, this printer uses patterns of dots to produce text and graphics. As a result, the Imagewriter can make anything the screen can display.

The Imagewriter actually comes in two models, but the only material difference between them is the width of paper they accommodate. The standard model holds paper up to 10 inches wide, while the wide-carriage model can handle 15-inch paper, which is handy for large spreadsheets. (The wide-carriage model will not be available until the second quarter of 1984.) See Table 1 for an overview of the Imagewriter's features.

Connecting an Imagewriter to a Macintosh is easy: Plug one cord into a grounded outlet, another into the printer socket on the back of the Mac, and slip in some paper. Switch on the printer and make sure its SEL light is glowing; the Imagewriter is ready to print.

Snapshots

Using various keystrokes, you can take "snapshots" of what is showing on the screen at any time. To get an instant print of the whole screen, press the Command key (⌘) and 4 key simultaneously. For a printed copy of the active window only, press Caps Lock, ⌘, and 4 together.

If you'd rather save the screen image and print it later, press ⌘ and 3 simultaneously; the whole screen image will be stored in a document named Screen 0. The second time you press ⌘-3, a current image of the screen goes into a document named Screen 1, the third time into a document named Screen 2, and so on up to Screen 9. To create more than ten snapshots, you must rename a current snapshot (such as Screen 0 to Screen 00), because the Mac will not accept a file that has a name already in use. All snapshot documents are standard MacPaint documents, so you can use MacPaint to enhance them before printing.

Screen snapshots have their uses, but most of the time you'll want to print whole documents, which rarely fit in a window. Each application has a Print command, usually located in the File menu. Choosing the Print command prints either the document you're currently viewing or another document whose name you specify.

Print Quality

The Imagewriter does a remarkable job of accurately reproducing the Mac screen's high-resolution graphics. The dimensions of a Mac screen and its printed counterpart are close enough for visual parity. Circles look round, squares print square, and text generally looks the same on the printer as on the screen (see Figure 1).

The printer actually has finer resolution than the screen. Horizontally, the printer can resolve 160 dots per inch, com-

With a good ribbon, printed images appear at least as dark, dense, and well-defined as displayed ones.

pared to the screen's 72. Vertically, the ratio is 144 to 72. The printer can overlap dots, but the screen cannot. With a good ribbon, printed images appear at least as dark, dense, and well-defined as displayed ones. One factor does detract from a printed image's sharpness and clarity: the Imagewriter uses a fabric ribbon, which cannot produce crisp dots on porous computer paper. Fuzzy edges on dots sometimes help fill unwanted gaps, but they can also make the printout look splotchy.

The screen displays images with square dots, while the Imagewriter prints them with round dots. Dot shape disparity causes some barely perceptible variations between displayed and printed images. Figure 2 compares a magnified image on the Mac screen to the same image printed on an Imagewriter.

The Imagewriter sometimes has trouble printing certain background patterns with complete regularity. The root of the problem is the pin-feed mechanism. While it is reliable (paper jams are rare), it is not as precise as the more elaborate tractor-feed mechanisms found on more expen-

The screen displays images with square dots, while the Imagewriter prints them with round dots.

sive printers. Paper can shift ever so slightly, especially lightweight (cheap) fanfold paper. The shift shows up as faint horizontal lines running across certain backgrounds, notably the standard gray desktop background (see Figure 3). Those ghostly lines never appear in plain text, with its solid, unpatterned characters, however. You can minimize this effect by using heavier fanfold paper or by forsaking pin-feed paper altogether in favor of single-sheet paper.

Text reproduction on the Imagewriter is excellent, and in most cases matches quite well what you see on the screen. Some combinations of type font, size, and style do look slightly different on the printer, however. The differences in dot size and shape between the printer and the screen, along with dot fuzziness from ink bleed, account for the aberrations.

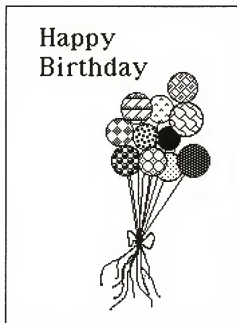


Figure 1
A comparison of screen and printer output. The dimensions of the printout closely match those of the screen display.

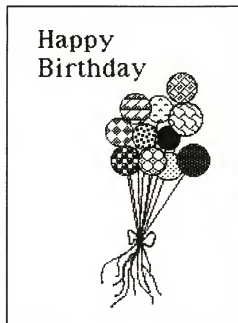


Figure 2
Magnified screen and printer output. Screen dots are square, while printer dots are round.

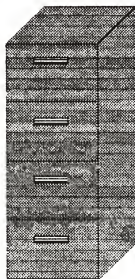
Printing Options

In most applications, a dialog box appears when you choose the Print command (see Figure 4). It lets you choose how much to print and the quality of the print resolution. Your choice of print quality—draft, standard, or high—determines how closely the printed image will match the displayed one.

Draft quality printing ignores pictures altogether and prints characters in one font, with no variation in style or size. Both standard and high quality printing reproduce pictures and text exactly as displayed. Standard quality images look exactly like those displayed on the screen, but high quality images have about twice as many dots, giving them a fuller, denser appearance. Figure 5 illustrates the differences among the three printer resolution qualities and the screen resolution quality.

The print resolution quality also affects how fast the Imagewriter prints. Draft quality is fastest, because it simply sends out a code for each character. The Imagewriter translates those codes into dot patterns, using its own built-in character font. The nominal print speed for draft quality is 120 characters per second.

Standard quality is somewhat slower than draft quality, because it sends dot patterns, not character codes, to the printer. The average print speed for standard quality is 90 seconds per page. High quality is slowest of all. It sends two sets of dot patterns for every line, so the Imagewriter has to make two passes across each line. The average print speed for high quality is 165 seconds per page.



Solid Oak File Cabinet

Replica of the one in grandfather's office. Matches our rolltop desk (see p. 21) and oak office chairs (see p. 19) for a complete set. Hand-rubbed tung oil finish in your choice of natural, light oak or dark stained oak.

FC602427L.....light oak.....\$195

FC602427D.....dark oak.....\$195

Shipping weight, 75 lbs.

Figure 3

Faint lines caused by paper shift. Quality single-sheet stationery minimizes this problem.

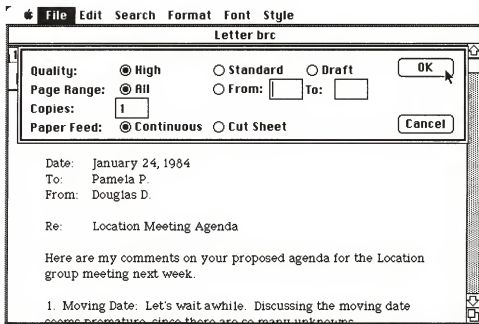


Figure 4

Printing options. This dialog box appears when you choose the Print command.

Printer Compatibility

The Imagewriter is manufactured for Apple by Tokyo Electric, the same company that makes C. Itoh printers. However, that does not mean the Imagewriter and the C. Itoh printers are interchangeable. Apple designed the Imagewriter's custom case to make it significantly quieter. That alone doesn't make other printers incompatible; it's the custom electronics inside that make it impossible to simply replace the Imagewriter with another printer.

Your choice of print quality determines how closely the printed image will match the displayed one.

The Mac uses a special program called a *printer driver* to send information to the Imagewriter in a form the printer's custom electronics can interpret. Other brands of dot matrix printers will not work with the standard Mac printer driver, nor will daisy wheel printers. But the standard printer driver is a computer program, and you can replace it with another program that drives another kind of printer. Apple plans to release the information that will allow programmers to write drivers for various printers.

Other Apple computers, such as the Apple II, Apple III, and Lisa, can also use the Imagewriter. All it takes is an appropriate adapter kit from Apple. Each kit includes a printer driver program, a special cable to connect the printer and the computer, and a reference manual.

Apple's Imagewriter Printer

What you see is what you get -- that's the watchword of the Apple Imagewriter printer. Like the Mac screen, this printer uses patterns of dots to represent text and graphics. As a result, the Imagewriter

Apple's Imagewriter Printer

What you see is what you get -- that's the watchword of the Apple Imagewriter printer. Like the Mac screen, this printer uses patterns of dots to represent text and graphics. As a result, the Imagewriter

Apple's Imagewriter Printer

What you see is what you get -- that's the watchword of the Apple Imagewriter printer. Like the Mac screen, this printer uses patterns of dots to represent text and graphics. As a result, the Imagewriter

Apple's Imagewriter Printer

What you see is what you get -- that's the watchword of the Apple Imagewriter printer. Like the Mac screen, this printer uses patterns of dots to represent text and graphics. As a result, the Imagewriter

Figure 5
A comparison of draft, standard, and high-resolution print qualities and the screen resolution quality.

Imagewriter at a Glance

Resolution	160 horizontal by 144 vertical dots per inch, maximum
Speed	1.5 minutes per page, standard quality, standard carriage 2.75 minutes per page, high quality, standard carriage 2.5 minutes per page, standard quality, wide carriage 4.5 minutes per page, high quality, wide carriage 120 characters per second, draft quality (text only)
Character size	Depends on application software, but typically 8- to 72-point, proportionally spaced
Line spacing	Depends on application software, but variable in increments of $\frac{1}{16}$ inch
Copy thickness	0.28 mm maximum (3 copies)
Paper width	10 inches, standard carriage 15 inches, wide carriage
Printing mechanism	9-wire printhead
Noise level	53 decibels, maximum
Paper feed	Single-sheet, pin-feed for continuous forms
Data connection	Mac serial port (RS-232 standard)
Power requirements	120 VAC, grounded 180 watts maximum, 16 watts standby 220 VAC models available
Size	18W × 12D × 6H inches, standard carriage 24W × 12D × 6H inches, wide carriage
Approximate weight	19 lbs., standard carriage 30 lbs., wide carriage

The Future

For now, the Imagewriter should satisfy most Mac users' needs for printed output. Other printer manufacturers will certainly write printer drivers, providing some competition, but it will be difficult to beat the Imagewriter's quality for the price. Those who cannot tolerate characters composed of dots will be willing to sacrifice multiple type fonts, sizes, and styles, along with the ability to print graphics, for their cherished typewriterlike characters. Daisy wheel printer manufacturers will no doubt accommodate them somehow. Perhaps the future lies somewhere in the realm of ink-jet or laser printer technology.

Apple Imagewriter Printer
Apple Computer, Inc.
20525 Mariani Ave.
Cupertino, CA 95014
800/538-9696; in California
800/662-9238

List price: \$495 standard carriage; price not set on wide-carriage model

Macware News

Announcements of new Macintosh products

Edited by Janet McCandless

Macware News reports on the latest hardware and software developed for the Macintosh. Products are listed by names of manufacturers, who provide information for the product descriptions. Prices and release dates are manufacturers' estimates. In this issue, we present a sample of products expected to be available immediately or in the near future; the list is by no means inclusive. Selected products will be evaluated in the Review section in this and future issues of Macworld.

Hardware

■ **Apple Computer**
20525 Mariami Ave.
Cupertino, CA 95014
800/538-9696; in
California 800/662-9238

AppleLine

An interface device that allows the Macintosh to be used as an IBM 3278-2 terminal for accessing information on large IBM mainframes. The self-contained unit connects directly to the 3274/3276 type-cluster controller coaxial cable and the Macintosh serial interface. The

product enables users to develop applications on a Macintosh for transfer to a mainframe, and vice versa. Available during the first quarter of 1984. List price unavailable at publication time.

Modem 300/Modem 1200

An intelligent modem (available in both 300 bps and 1200 bps versions) packaged with a DemoPak from CompuServe, a Premium Pak from The Source, and an Accessory Kit that contains the cables and manuals necessary for installation. Both

units through simple commands. Available upon release of the Macintosh. List price: \$225 Modem 300, \$495 Modem 1200.



External Disk Drive, Apple Computer, Inc.

External Disk Drive

An additional disk drive that has a memory capacity of 400K and connects directly to the drive port. The drive is enclosed in a 6- by 8-inch plastic housing and uses 3½-inch hard shell micro-floppy disks. Available upon release of the Macintosh. List price: \$495.

Carrying Case

A padded carrying case that holds the main unit and has inside pockets fitted for the keyboard, mouse, manuals, and power cord. The pack is made of Cordura nylon, which is both water and tear resistant. Designed to be carried either by

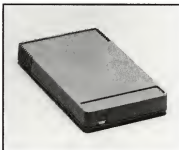
hand or on the shoulder (a shoulder strap is included), the case is padded to protect the Mac system during transportation in a car or plane; however, a more protective container is recommended for shipping. The case fits under the seats of most commercial airlines. The Mac and carrying case together weigh 22 pounds. Available upon release of the Macintosh. List price: \$99.



Carrying Case, Apple Computer, Inc.

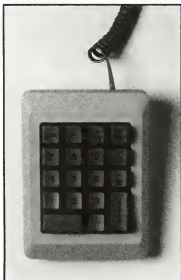
Keypad

A ten-key numeric digital keypad that connects directly to both the Macintosh keyboard and the main unit. The keypad contains its own microprocessor. It can be used alone with the main unit and can be



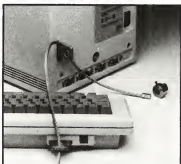
Modem 1200, Apple Computer, Inc.

modems are equipped with an RS-232C interface that has a DB-9 connector. The modems feature self-diagnostics and operate with MacTerminal communications software. Built-in intelligence allows auto-dial (touch-tone or pulse dialing) and auto-answer, and enables users to customize these func-



Keypad, Apple Computer, Inc.

operated with four system command keys. The unit weighs two pounds and measures 6 by 4 inches. Available upon release of the Macintosh. List price: \$99.



Security Kit, Apple Computer, Inc.

Security Kit

A kit that allows the keyboard and main unit to be locked to protect the Macintosh from theft. One keyhole lock slips into the keyboard and the other into the main unit. A steel cable is put through the two locks. The end of the cable is inserted into a metal cylinder that has several tamper-proof screws. When screwed in with the kit's special screwdriver, they lock the cable. For complete security, a padlock also may be attached. Available upon release of the Macintosh. List price: \$49.

3½" Disk Box

Micro-floppy disks encased in hard shells for use with the Macintosh's Sony disk drives. The disks measure 3½ inches and are durable and compact. The exterior shell and metal shutter protect the media from damage and data loss due to improper handling or dust. Available upon release of the Macintosh. List price: \$49 for 10 disks.



3½" Disk Box, 10 micro-floppy disks, Apple Computer, Inc.

■ Davong Systems 217 Humboldt Ct. Sunnyvale, CA 94086 408/734-4900

The following products will be available during the first quarter of 1984. List prices were unavailable at publication time.

Davong Universal Winchester Disk System

A single-drive hard disk system available in 5-, 10-, 15-, 21-, and 32-megabyte (M) formatted capacities. The system supports one master hard disk, one slave hard disk, and one tape drive. Data can be backed up with a

28M streaming cartridge tape drive. Average access time for 5M to 21M systems is 110 milliseconds, 40 milliseconds for 32M systems. The minimal transfer rate is 62.5 kilobytes per second. The Winchester runs on the Macintosh operating system and provides volume backup and disk cache.

Davong Streaming Tape Backup

A backup drive that copies data from the Winchester Disk System or from floppies onto ¼-inch tape cassettes. The streaming tape has a formatted capacity of 28M. The system copies data in buffered stop/start or streaming mode and backs up 15M in approximately 4 minutes. A cyclical redundancy check procedure screens data for errors.

Davong Multilink Local Area Network

A shared, centralized hard disk storage system that uses a synchronous data link communications (SDLC) loop and operates with Z8030-based protocol. The stations, or nodes, are linked by conventional coaxial cable (RG-62, IBM3270) to a hub device. The network can

connect up to 255 nodes. Each node uses a network multifunction card that supports external hard disk and network interfacing, network boot features, and optional I/O expansion. Data is transferred at 1 megabit per second, and messages can vary from 1 to 540 bytes in length. Davong has developed *Mac Net* networking software for the system and has designed optional software, *Mac OS*, that moves files between the Macintosh and the IBM PC. Other Multilink options include print spooling and electronic mail.

■ Prometheus Products, Inc.
45277 Fremont Blvd.
Fremont, CA 94538
415/490-2370

Promodem 1200

A 300/1200 bps stand-alone modem compatible with Bell 212A- and 103-type modems. Standard features include auto-originate/auto-answer, an auto-

modem commands. During a power failure, the Options Processor battery retains real clock time and date and restores the default operating parameters of the modem. A business/personal directory stores telephone numbers; directory entries may be assigned individual or group reference names. A receive/transmit buffer stores messages, which can be sent later to numbers selected by their individual or group reference name. Available upon release of the Macintosh. List price: \$99.

■ Tecmar Personal Computer Products Division
6225 Cochran Rd.
Cleveland, OH 44139
216/349-0600

The following products are available upon release of the Macintosh. List prices were unavailable at publication time.

Removable Cartridge Winchester

A hard disk that has a 5-megabyte storage capacity and comes with a removable cartridge. The unit combines the advantages of removable, transportable media with the assets of high-density, high-speed information access.

IEEE 488 Interface

An industrial standard interface that enables transfer of data from general purpose interface bus peripherals to the Macintosh via Applebus. The unit connects to the Applebus port and enables the Macintosh to function in a variety of industrial applications including sophisticated measure instrumentation, chromatography, and process and industrial control. The board has fully implemented IEEE 488 talker, listener, and controller functions. It contains its own power supply.

Telephone/Modem

A telephone and modem interface that offers a cohesive approach to communications. The board is mounted in an external cabinet and functions as a standard telephone with a handset, or it can be connected to another telephone. A demodulated touch-tone decoder detects incoming calls and prevents piercing sounds or garbling while the phone is being used. The unit generates a phone log without intervening on the line. The modem is 300/1200 bps (212A compatible) and has a pulse-tone automatic dialer. Dual-tone multiple frequency allows users to interface any audio source (such as a speech synthesizer) to the telephone.

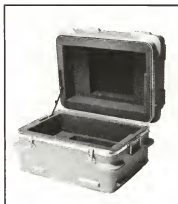
Print Buffer

A buffer that permits users to employ the computer's full functions while printing text. It is available in 16K and 64K versions and has its own power supply.

■ Thermodyne International
20850 S. Alameda St.
Long Beach, CA 90810
213/603-1976

Shok-Stop

A polyethylene carrying case that has a foam core designed for cargo transportation. The case is certified for shipment by airlines. Channels and corner bumpers protect the contents under high-stress conditions. The case, which measures 22 by 18 inches with a 12-inch bottom and a 5-inch top, has recessed hinges and latches, spring-loaded handles, and interlocking ribs for stacking. Its materials resist corrosion and fungus. Available upon release of the Macintosh. List price: \$230.



Shok-Stop, Thermodyne International

■ Verbatim Corporation
323 Soquel Way
Sunnyvale, CA 94086
408/245-4400

Verbatim Datalife Microdisk

A pocket-sized 3½-inch microdisk with a storage capacity equivalent to that of the 5¼-inch minidisk. A shutter door and plastic shell protect the media. The microdisks are manufactured with a write-protect mechanism and a metal



Promodem 1200, Prometheus Products, Inc.

dialer, tone and pulse dialing, an RS-232 interface, a real-time clock/calendar, a programmable intelligent dialing capacity, alphanumeric display, internal and remote diagnostics modes, help commands, an internal power supply, and a built-in speaker with volume control. Available upon release of the Macintosh. List price: \$495.

Promodem's Options Processor (sold separately) adds features that may be used independently or in combination with the auto-dialer and other



Datalife Microdisk, Verbatim Corporation

hub. Disks are packaged in boxes of five, and twinpacks and boxes of ten microdisks will be marketed later. Available upon release of the Macintosh. List price: less than \$6 per disk.

Software

■ **Apple Computer**
20525 Mariani Ave.
Cupertino, CA 95014
800/538-9696; in
California 800/662-9238

Macintosh Assembler/ Debugger

A program that features a mouse-based editor, a 68000 macro assembler, and a symbolic debugger. The editor handles multiple files concurrently (file sizes are limited only by disk space) and follows user interface conventions. The macro assembler produces relocatable object code and symbol files for symbolic debugging.

The mouse-driven symbolic debugger can be operated remotely from a second Macintosh—a significant benefit when the machine is being debugged since the program uses less space. Multiple windows display memory contents in various formats including instruction and symbolic disassembly. Available during the first quarter of 1984. List price: \$99.

MacDraw

A structured graphics editor that enables users to prepare flow charts, graphs, technical drawings, organizational charts, and freehand drawings. The program also allows users to move, change, or transfer graphics to other applications. These flexible functions aid in redesigning presentation graphics or in adjusting blueprints.

As with *MacPaint*, users can select several predefined shapes including arcs, rectangles, squares, and circles. Text can be added using various type fonts, styles, and sizes. Users can work on up to four documents simultaneously. Available during the second quarter of 1984. List price: \$125.

MacProject

A program for scheduling and illustrating project development that charts and displays resource usage, tracks costs, and graphically demonstrates relationships between activities. Users can define tasks, indicate the time each will take, draw relational lines between jobs, and enter their start dates. The program then charts a critical path and determines the project's completion date.

Conversely, it can also calculate the necessary start date if a completion date is known. Users can transfer data from spreadsheets to *MacProject* documents. Combinations of activities can be selected and duplicated in other locations on the chart. Available during the second quarter of 1984. List price: \$125.

MacTerminal

A program used for asynchronous communications that emulates the VT-100 and VT-52. By emulating the VT-100, the Macintosh can receive commands from timesharing services to position the cursor, display information, and skip a designated number of characters or lines.

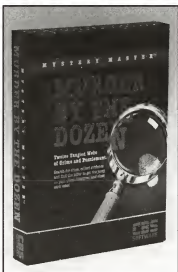
MacTerminal sends and receives files at speeds of up to 19,200 bps. Users can store data and scroll back through the data. Text can be cut and pasted to the terminal from other applications, and vice versa. Dialog boxes and forms simplify the steps in configuring the program to communicate with other terminals. These configurations can be saved in a file for future reference. Available during the first quarter of 1984. List price: \$99.

Macintosh Logo

An interactive programming language with graphics output, designed for children. Children draw pictures by learning how to move a turtle on screen. The language helps to build programming skills; by giving commands to the turtle the child learns how to draw a square, and by grouping commands, how to make a statement. Available during the second quarter of 1984. List price: \$99.

CBS Software

One Fawcett Pl.
Greenwich, CT 06836
203/622-2615



Murder by the Dozen, CBS Software

Murder by the Dozen

A game that offers players 12 murder cases to solve. Information supplied by the crime computer may be used to obtain a case history of the victim, detect physical evidence, travel to various locations, and interview suspects. The game is won by the person who identifies the criminal and the motive in the shortest time. Up to four detectives can play. The game

package includes two reference manuals with clues and solutions, mystery master worksheets, and a solution decoder. Available during the first half of 1984. List price unavailable at publication time.

■ **Chang Labs**
5300 Stevens Creek
Blvd. #200
San Jose, CA 95129
408/246-8020

The following products will be available during the first or second quarter of 1984. List prices were unavailable at publication time.

MicroPlan

A financial modeling program with built-in formulas. *MicroPlan* provides a complete statistical package, including formulas for amortizing monthly loans, standard deviations, and tax rate schedules, and for determining the impact of loan structure on cash flow. It offers a help command and built-in prompts. Its worksheet size extends to 200 rows and 60 columns (approximately 1000 entries per table). The report generator includes dollar signs, commas, negative numbers, percent signs, underlining, centered titles, and subtitles.

Consolidated Module

A program to retrieve and consolidate data from different files and perform calculations, variances, and sensitivity analyses. The program can merge tables and add or subtract reports from different departments. Used with *MicroPlan*, its interrelated reports can be revised to build up-to-date management summaries.

Link Module

A *MicroPlan* spreadsheet option for analyzing data from mainframe files, timesharing services, and files from other application and spreadsheet packages. With the *Link Module*, *MicroPlan* tables can be read by data base and word processing applications. Built-in routines post records to individual or multiple rows or columns and cross-tabulate and analyze results.

FilePlan

A data base program that uses a spreadsheet format for data entry. Records contain between 128 and 1024 characters, with a total of 32 variable-length fields (each using up to 99 characters). Users may view as many as 15 records at a time, sort data on 5 fields, and apply the query feature to extract information. Field and record parameters may be defined after data entry. The program includes prompts that can be customized.

GraphPlan

An integrated business-graphics spreadsheet package with built-in formulas, statistical commands, and graphics, sorting, and ranking capabilities. The program features automatic generation of legends, numerical, date, time, and logarithmic X- and Y-axis labels and tic marks. Its graphics include explodable pie charts and horizontal or vertical line and bar graphs.

MemoPlan

A word processing program oriented for memo and letter writing. Users may work on up to five documents simultaneously, flipping between files with a single keystroke. The program has been designed to save and recover documents automatically in the event of a power failure. It offers standard word processing features (such as copy, move, delete, insert, justify, onscreen underline, split screen, and search and replace), but does not include headers, footers, or page-number print options.

DocuPlan

A word processing program designed to format long reports that require footnotes, subscripts/superscripts, indexes, and/or tables of contents. It has the capability to prepare itemized lists with bullets or numbers, and adjust line spacing, indentation, and margin changes.

■ Desktop Software Corporation

228 Alexander St.
Princeton, NJ 08540
609/924-7111

NPL Information Management System

A nonprogrammer's language for application development that uses simple sentences to collect data, make ad hoc queries, and generate reports. The language is similar to the popular data base query languages RAMIS II, FOCUS, and NOMADZ. Users can create data bases; specify edit and validation tests; select, sort, and merge data records; and print

reports in tabular or letter format. Each record contains up to 4000 bytes and 100 fields. Up to 100 fields may be indexed using up to 10 sort keys and 30 verb objects per request.

The program is for use by businesspeople and professionals who design specific applications software and by firms whose managers and researchers extract and summarize information from large files and interact with corporate mainframes. The package includes a query primer, a data entry primer, a user's guide, and a system disk with sample programs and data files. Available during the first half of 1984. List price: \$950.

■ Dilithium Software

8285 S.W. Nimbus #151
Beaverton, OR 97005
800/547-1842; 503/646-2713

Telofacts 2

A software package that assists users in gathering and analyzing information and designing questionnaires. The program demonstrates designing a questionnaire for data obtained from respondents (statistics, resume information, marketing or survey data) and entering the answers, and presents choices for analyzing and displaying results. Questionnaire results can be displayed for one question or an entire survey in

report format, ranks, lists, or scores. The package includes an installation guide, documentation, a reference card, and a disk. Available during the first quarter of 1984. List price: \$199.95.

■ Living Videotext

1000 Elwell Ct.
Palo Alto, CA 94303
415/964-6300

Think Tank

A program that offers users flexibility in recording and reorganizing ideas. The program's outline format enables users to expand or eliminate subheads and restructure sections without affecting overall text structure. The program exploits use of the mouse and the high-resolution graphics screen. It controls the amount of text to be viewed, searches for key words, stores data on screen, reorganizes or restructures data, and prints. Available during the second quarter of 1984. List price unavailable at publication time.

■ Microsoft Corporation

10700 Northup Way
Bellevue, WA 98004
206/828-8080

Microsoft Budget

A program that helps users construct operating budgets and variance analysis worksheets. The budget planning process, based on the variable costing method, develops six interrelated *Multiplan* worksheets to build an operating budget. The

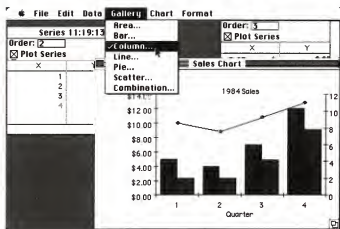
program helps to calculate the costs of running a business. You can create worksheets to calculate manufacturing overhead, unit costs, costs of goods sold, and sales and administrative expenses, as well as sales budget worksheets to project revenues.

The Operating Budget worksheet copies data from other worksheets and calculates the contribution margin and net operating income for products, and can demonstrate the effects of changes in unit sales volume.

The Variance Analysis worksheet is a budget control tool divided into three sections: the budget variance summary shows variances between actual performance and results forecasted in the Operating Budget worksheet; detailed variance analysis shows variances for sales volume, production costs, fixed overhead, and sales and administrative expenses; and worksheet data provides a matrix in which to enter data used to recalculate the summary and detail sections of the Variance Analysis worksheet. Available during the second quarter of 1984. List price: \$100.

Microsoft Cash Budget

A program that monitors cash flow to provide managers with accurate data for marketing, finance, and sales decisions. *Cash Budget* creates five *Multiplan* worksheets to detail cash flow and show net cash flow. The first four determine cash inflows from sales, cash outflows for purchase of materials and merchandise, cash flows from operations, and non-operating cash flows. *Cash*



Microsoft Chart, Microsoft Corp.

Budget then consolidates those results to produce a net cash flow position for each interval selected. The final result works as a planning and strategy tool. Available during the second quarter of 1984. List price: \$100.

Microsoft Chart

A program that translates numeric data into graphic form to illustrate relationships or comparisons presented by the data. Users assign information to a set of categories; each category has a corresponding value. The category-value sets form data points, which may be combined to form up to 64 data series.

Once data has been assembled, users choose a basic chart type from the Gallery menu, depending on their objectives for presenting the data, the device on which the chart will be printed or displayed, and the amount or type of data included. The program allows for changes in shadings, colors, line thicknesses, and positions of titles. Shading and line format options are presented in dialog boxes.

The mouse is used to select objects on the chart and move them to different positions. Text placement on charts is user controlled. A group of

charts can be created on screen, or two or more charts can be overlaid. Charts automatically reflect changes to data series. Available during the first quarter of 1984. List price: under \$200.

Microsoft File

A nonprocedural tool that helps individuals maintain personal data files. The program offers forms-based data entry and retrieval. Users select a format for data entry suited to their personal file management needs from a gallery of standard data forms. Users can adapt data forms by determining the number of fields, field sizes, and field designations, and by selecting data orientation by columns or rows.

File's reporting capabilities include sorting and summing on any field, simple mathematical operations, averages, counts, and logical operators. The interactive program permits users to transfer data entered in it to other applications. Available during the first quarter of 1984. List price: under \$200.

Microsoft Financial Statement

An evaluation tool that assesses a company's financial performance. The program helps users create balance sheets and income statements based on periods (e.g., quarters or years) they select. Using worksheet data, *Financial Statement* automatically calculates 18 financial performance ratios used by accountants and financial professionals. The ratios are divided into four categories: liquidity, leverage, profitability, and efficiency. Users are then able to gauge their firm's ability to service current and long-term debts, to measure net returns, and to evaluate how efficiently company resources are being used. Available during the second quarter of 1984. List price: \$100.

Microsoft Personal Finance

An interactive analysis tool for personal financial planning that addresses cash flow analysis. Users can produce a financial forecast through the development of four worksheets. The main worksheet summarizes all sources and uses of cash and includes a personal balance sheet. The three supplementary worksheets can be used to detail earnings, living expenses, assets, and liabilities. They can be linked to the main worksheet for analysis. Cash flow analysis enables users to plan net worth positions with variable net cash positions. Available during the second quarter of 1984. List price: \$100.

Microsoft Word

A word processing program that uses the mouse, windows, dialog boxes, and graphics to provide a wide range of text editing capabilities. The document window uses the standard Macintosh scroll bars, split bars, and size box. Moving or copying text and graphics is accomplished by standard Macintosh editing functions.

The program displays on-screen character formats (e.g., bold, underline, or italic) and character positioning. List boxes offer choices in font and type sizes. Character styles can also be selected from the keyboard, using the ⌘ (Command) key. Additional features allow searching to a page number, footnote placing and sequencing, customizing of style sheets, compiling of glossaries, automatic paragraph reformatting, and paragraph and page formatting. Available during the first quarter of 1984. List price: under \$200.

Peat Marwick Mitchell & Co.

345 Park Ave.
New York, NY 10154
212/758-9700

SEACAS

A microcomputer-based tool for automating auditing tasks. When fully implemented, the program is expected to reduce total audit hours by at least 10 percent and to improve audit documentation. SEACAS (Systems Evaluation Approach/Computerized Audit Support)

facilitates the development of audit control programs, and assists auditors in posting audit adjustments and accurately updating trial balances and financial statements.

In conjunction with the audit retrieval package System 2190, data to prepare financial models can be retrieved from client files and public and central office data bases. Clients need not own a computer to use the program. Available during the first or second quarter of 1984. List price unavailable at publication time.

Software Publishing Company

1901 Landings Dr.
Mountain View, CA 94043
415/962-8910

PFS File

A data base program that permits users to custom design forms that can be modified or updated. PFS File offers the basic data base management features (recording, storing, updating, and sorting information), but simplifies entry and sort features for first-time computer users. Field attributes do not have to be defined, and items can be multiline. Common applications for the forms include invoices, personnel records, book indexes, and real estate listings. Available during the first half of 1984. List price: \$100.

PFS Report

A program that produces tabular summaries for presentation-quality reports from data stored in PFS File. It sorts alphabetically, numerically, or by key words and calculates averages, counts, and totals. Users

can format data to center, underline, and align columns, and number pages. Available during the first half of 1984. List price: \$100.

Both *File* and *Report* use the Macintosh's specially designed features. With the mouse, users can quickly move to any position on the screen or within a PFS form. Menu bars and pull-down menus let users select functions at any time. Functions can also be selected by pressing one or more keys. Windows enable users to view the form at all times, regardless of which function is being performed on it.

Trade*Plus

460 California Ave.
Palo Alto, CA 94306
800/952-9900;
in California 800/972-9900;
415/324-4554

A service that gives investors last sale prices and stock market data. Subscribers can trade all stocks and stock options listed in the *Wall Street Journal* directly through their personal computers. Stock and option prices can be transferred into worksheets, and users can obtain up to three months of historical data formatted in ASCII, DIF, or *Multiplan*. Separate IRA and Keogh accounts may be maintained, and users can trade securities in these accounts as well.

*Trade*Plus* tracks gains and losses for up to three portfolios and supplies tax records for each. The system is open to subscribers 24 hours a day, 7 days a week. Its release is based on the availability of Macintosh communications software. List price: \$195; 10 cents per minute, night; 40 cents per minute, day.

Volition Systems

P.O. Box 1236
Del Mar, CA 92014
619/481-2286

Modula-2

A language integrated with the Macintosh operating system that works on text files and generates code files that can be executed in the Macintosh working environment. The Modula-2 system combines compiler, module library, and library manager. Library modules include random-access files, disk directory operations, format conversions, strings, decimal arithmetic, storage management, program scheduling, and process scheduling. Developed by Niklaus Wirth, the creator of Pascal, the package contains sample and demonstration source programs (including p-shell and 40 tutorial source programs) and complete documentation. Available during the second quarter of 1984. List price: \$495.

Macware News uses information provided by manufacturers; it does not evaluate products or corroborate manufacturers' claims. Send a description and photograph of your new product to Macware News, Macworld, 555 De Haro St., San Francisco, CA 94107, 415/861-3861.

THE MOST USEFUL IDEAS ARE OFTEN REMARKABLY SIMPLE.



The first filing system was a remarkably simple way to do incredibly useful things. Like organize a business and make it work.

PFS:FILE and PFS:REPORT are remarkably simple software programs that can help you organize your business faster and more effectively. And both programs are amazingly easy to learn.

FILE and REPORT were designed for people without computer experience. So all instructions are easy to understand. And you can be productive quickly.

FILE lets you record, retrieve and review information like personnel records

and purchase orders. Instantly. You can even cross reference by department or job or however you want. And you arrange information in "forms" you design yourself. So your filing system will fit your needs. Not your computer's.

REPORT is the perfect companion program to FILE. Because it gives you the power to summarize and perform calculations on information you've stored with FILE.

With REPORT, you can create presentation quality summaries in table form in just minutes. Automatically sorted, calculated, formatted and printed!

FILE and REPORT are both fully integrated with the PFS Family of Software, including PFS SOLUTIONS which are pre-designed forms and reports for popular applications.




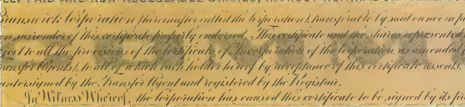
PFS: It's the powerful software that's simple to learn. It's the power of simplicity.

See your computer dealer for more details.



PFS SOFTWARE THE POWER OF SIMPLICITY

The PFS Family of Software currently runs on IBM®, Apple®, Radio Shack, Digital, Hewlett-Packard, Texas Instruments, Panasonic and other personal computers. © Software Publishing Corporation.

		por	
		2	3
1			
2			
3			
4			
5			
6			
7			
8			
9			
10			
11			
12			
13			
14			

Portfolio

4

THIS CERTIFICATE IS
TRANSFERABLE IN
NEW YORK OR CHICAGO
OR SKOKIE, ILLINOIS

Investing with Multiplan

Andrew T. Williams

The best way to find out what a program really can do is to make it work. When you put Multiplan to the test, you'll be pleasantly surprised. Just use the mouse to move the pointer to the cell in which you want to make an entry, click the mouse button, and you're ready to type text or a number, paste in a formula, or create your own function. Multiplan implements the pull-down menus, dialog boxes, scroll bars, and other unique features of the Macintosh's advanced technology.

Constructing a simple stock portfolio valuation worksheet with *Multiplan* demonstrates the program's powerful formula-creating and formatting features. This sample worksheet is set up to compute the worth of investments on a daily basis. To begin constructing it, set all column widths to 8 characters. At that width, the 9 columns of the portfolio valuation worksheet will appear on the screen at one time, simplifying the task. To set the column width, pull down the Select menu and choose the All option. Next, pull down the Format menu, choose the Column Width option, and set the column width to 8 characters.

Now you are ready to enter the headings shown in Figure 1. Begin by positioning the pointer over cell R1C3 and clicking the mouse button to select that cell. Then type **Portfolio Valuation Worksheet**.

While typing, you'll notice two important features of the program. First, *Multiplan* is smart. It knows you are typing text and accepts it without making you press an "alpha" key or type a special character, such as the quotation mark, to initiate a text entry. Secondly, all of the text you type will be displayed, even though it extends beyond the right boundary of cell R1C3. *Multiplan* has "soft cell boundaries," so that lengthy text entries can extend into adjacent cells, provided those cells are empty. These two features make entering text for titles, row labels, and column headings extremely easy.

After you enter the title, select cell R3C1 and enter the word **Date:**. While you still have R3C1 selected, pull down the Format menu and choose the Align Right option. Now select the block of cells from R5C1 to R6C9 for the column headings. The Mac gives you several methods for selecting this range. The easiest is to click the cell at one corner of the range, R5C1, and then move the pointer to the opposite corner, R6C9. When the pointer is on the cell, press the Shift key and click the mouse button. The entire range will be selected.

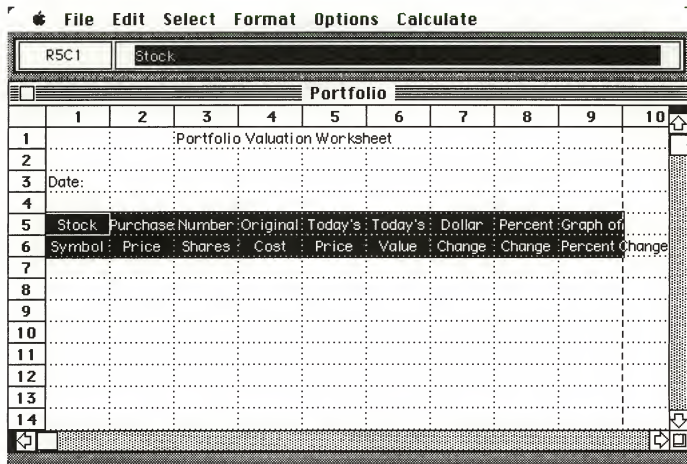


Figure 1
Portfolio valuation worksheet headings.

*Multiplan has
“soft cell boundaries,”
so that lengthy text en-
tries can extend into
adjacent cells.*

As with all selected cells or ranges, this one is displayed in reverse video (white characters on a black background). The first cell of the range, R5C1, has the solid white border that indicates it is the active cell. Your typing appears in this cell on the worksheet. Now type the word **Stock**. When you are finished, press the Enter key and watch what happens.

The range is still selected, but the active cell moves one cell to the right. Each time you complete an entry by pressing Enter, *Multiplan* advances the active cell so that the worksheet is ready to receive your next entry. You can skip over cells by repeatedly pressing the Enter key; and you can back up by holding down the Shift key when you press Enter. Don't use the Return key to finish an entry; it makes the entry, but it also cancels the range selection.

After you have entered the column headings shown in Figure 1 and while you still have the block selected, you can align the labels. *Multiplan* offers three choices for aligning text in a cell: left, right, or center. Column headings look best aligned either at the right or at the center of each cell. Pull down the Format

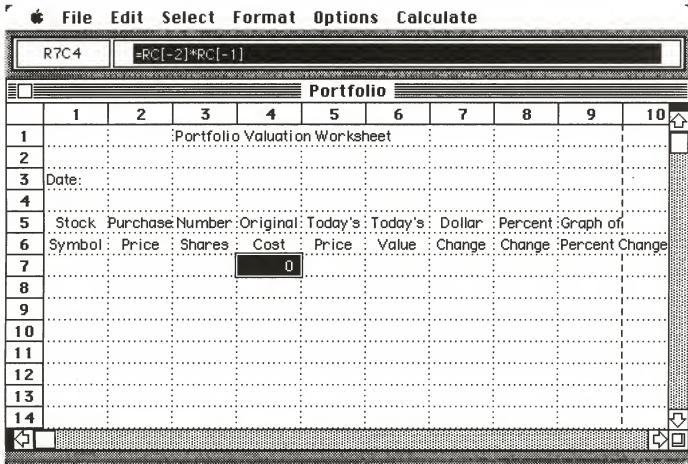


Figure 2

Entering the Original Cost formula into cell R7C4. The formula bar at the top of the screen shows the formula that is entered into the active cell.

menu, make your choice, and all the cells in the selected area will be aligned (in Figure 1 the labels are centered).

You are now ready to construct the formulas in the main body of the worksheet.

Formulas

A great advantage of electronic spreadsheets is that they allow you to create formulas to do routine calculations. This function facilitates changing a number to update a worksheet or to see what will happen if one of the underlying assumptions changes. You can always be sure the results are mathematically correct because they are calculated by the computer.

This portfolio valuation worksheet uses several formulas. The first is the Original Cost in column 4, which is the Purchase Price from column 2 multiplied by the Number of Shares from column 3. In a similar fashion, Today's Value, Dollar Change, and Percent Change can be calculated by properly defined formulas.

Defining a *Multiplan* formula is easy because you can use the mouse to point to the different cells. To create the formula for Original Cost, for example, move the pointer to cell R7C4 and select the cell by clicking the mouse button. Now type an equals sign (=) to announce to *Multiplan* that you want to create a formula. The equals sign appears in both the cell and the formula bar at the top of the screen.

Pointing with the mouse pays off here. Cell R7C2 contains the purchase price for the first stock, and that price will be the first part of the formula. Move the pointer to R7C2 and click the mouse button. The reference RC[-2] appears in the active cell and the formula bar. This designation refers to a cell that is relative to the active cell R7C4. It is *Multiplan's* way of saying, "Get the information from a cell in this row (R without a number means the current row), two columns to the left of the active cell"—C[-2].

Since you want to multiply the Purchase Price, RC[-2], by the Number of Shares from column 3, R7C3, type an asterisk (*). This symbol is the multiplication sign. Now click cell R7C3, and the formula bar will show =RC[-2]*RC[-1].

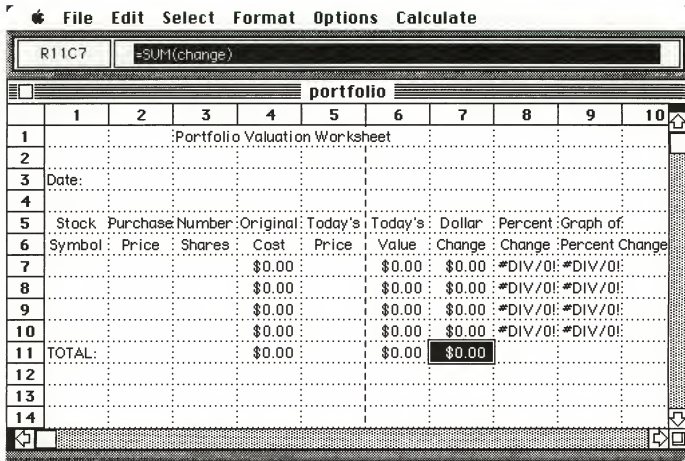


Figure 3

Sample worksheet with all formulas in place. Blank cells are for data entry. Cells containing formulas display either \$0.00 or #DIV/0!, which indicates a formula where division

by zero is specified. Once data is entered into the worksheet, a #DIV/0! will be replaced by a value.

Defining a Multi-plan formula is easy because you can use the mouse to point to the different cells.

The Original Cost formula is complete. Press Enter and the formula will be entered into cell R7C4. A zero appears in the cell because both the references in the formula are to cells that are, for the moment, blank (see Figure 2). When information is entered, the result will be calculated automatically.

Use the mouse to point to the appropriate cells to create the formulas for Today's Value (Today's Price times Number of Shares), Dollar Change (Today's Value minus Original Cost), and Percent Change (Dollar Change divided by Original Cost).

When you specify the Percent Change formula, #DIV/0! appears in cell R7C8. Don't worry; you haven't done anything wrong. *Multiplan* is just saying that you have created a formula that calls for the impossible: division by zero. As soon as data is entered so that cell R7C4, Original Cost, no longer contains a zero, the value calculated by the formula will replace #DIV/0!.

Don't bother to multiply the Percent Change formula by 100 to scale the decimal up to a percentage. The decimals will be displayed as percents (complete with percent signs) when you format the cells with the Percent option.

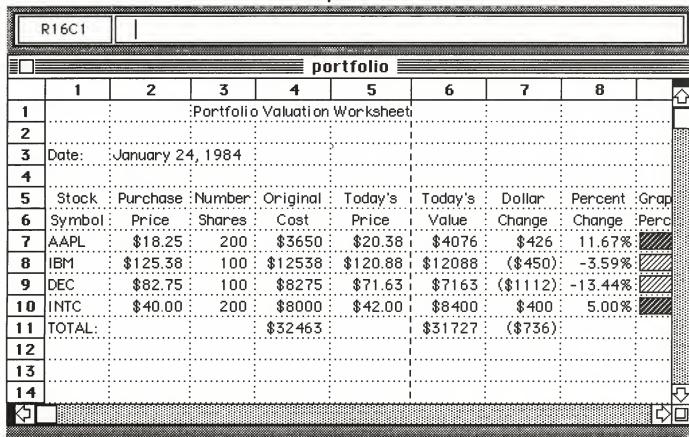


Figure 4
Completed sample worksheet with bar graphs in column 9.

You now have created the formulas for the first eight cells in the first row of the table. To complete the formulas, you need only to enter the formula for the Graph of Percent Change in column 9.

Bar Charts

Multiplan is not a graphics program, but it does use the Mac's high-resolution screen to draw excellent horizontal bar charts. The bars are either black (positive values) or white (negative values). Bars are drawn using the Bar Graph option from the Format menu. Each bar is one cell high, and its length is determined by the value of the number in the cell containing the bar.

To set the Bar Graph format, select cell R7C9, pull down the Format menu, and choose Bar Graph. Next, hold down the mouse button and drag the pointer up to highlight cells R5C9, R6C9, and R7C9. Cells R5C9 and R6C9 contain the words "Graph of Percent Change." Use the Left Align command from the Format menu to align these three cells to the left. The Left Align command displays the bars directly below the headings and next to the corresponding percentages in column 8.

Use the pointer to enter the formula **Percent change** • 100 into cell R7C9. Multiplying by 100 is used in this instance to produce easily distinguished bars. If your bars don't show the differences in percentage changes clearly, try another scaling factor such as 75 or 125.

Once again the message #DIV/0! appears, indicating that you have created a formula referring to a cell in which division by zero has been attempted. Everything will be fine when data is entered into the table.

Formatting Features

Multiplan has a rich assortment of format options with which you can improve the appearance of worksheets. You have already used an align command to line up the column headings. Now set the Dollar format to improve the appearance of the columns containing currency—columns 2, 4, 5, 6, and 7. Also set the Percent format for column 8 so the numbers in that column will display as percentages.

To set a format, first select the range to be formatted. Since you will be formatting entire columns, it is

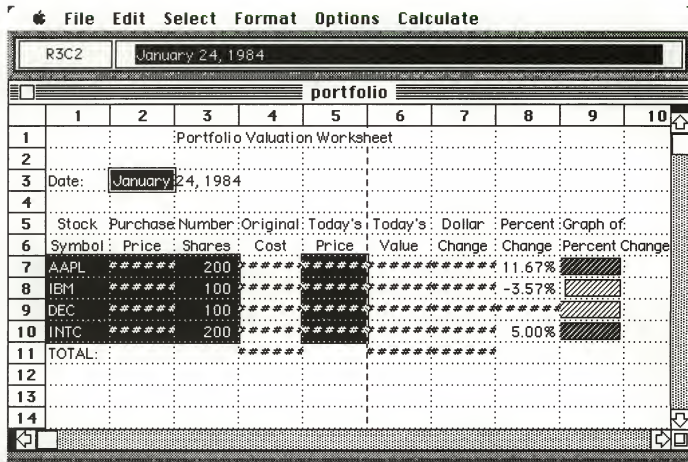


Figure 5

The sample worksheet filled with crossbatches, indicating that some columns are too narrow to accommodate data.

Multiplan uses the Mac's high-resolution screen to draw excellent horizontal bar charts.

easiest to select an entire column by clicking the mouse button over the column number in the border above the worksheet.

Since the columns containing currency aren't all next to one another, use the ⌘ (Command) key and the mouse button together to select a range of nonadjacent columns. Hold down the ⌘ key, point to a column, and click the mouse button. Do this to select columns 2, 4, 5, 6, and 7. Then pull down the Format menu and choose the Dollar option. Finally, click the column number for column 8 and set the Percent format.

Filling Up the Table

Now you are ready to use the Fill Down command to copy the formulas from row 7 down the table. But first you must select the area to be filled. Begin by placing the pointer on cell R7C1 and clicking the mouse button.

Decide how many stocks to put on the worksheet. In the example there are four stocks, so move the pointer to row 10 in column 9. This is the opposite cor-

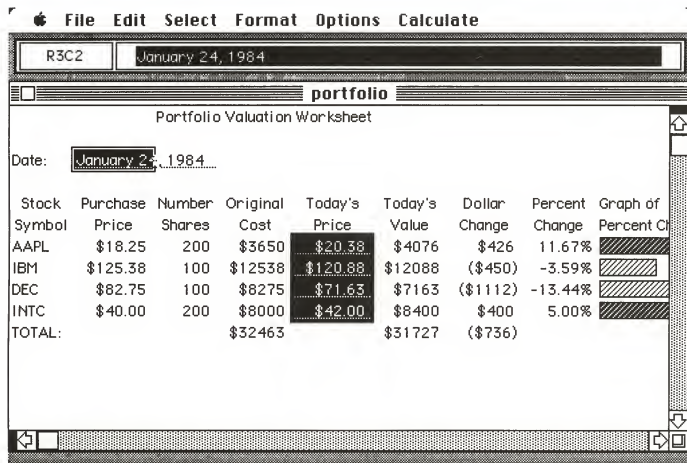


Figure 6

Worksheet after Protect Document command is executed. The unprotected cells are underlined and the active cell will move

from unprotected cell to unprotected cell each time you press Enter.

ner of the block formed by cell R7C1 and R10C9. With the pointer on R10C9, hold down the Shift key and click the mouse button to select the block.

Next, choose the Fill Down command from the Edit menu. Everything you specified for the cells of the first row will be copied down the respective columns to fill the rest of the cells in the range.

Totaling Your Position

You can total the Original Cost, Today's Value, and Dollar Change columns by placing the *Multiplan* function SUM in the appropriate cells along the bottom of the table. The SUM function adds the contents of a range of cells, and you can select that range either by pointing or by naming the range.

Begin by entering the word **TOTAL**: into cell R11C1. To see how the name feature works, name the range for Original Cost. Locate the pointer on cell R7C4, hold down the mouse button, and drag the pointer down to cell R10C4 to select the range to be named. Pull down the Select menu and choose the Define Name option.

A dialog box appears, and you can type in the name. This group of cells contains the Original Cost for each stock, so name it **COST**.

To sum the cells in COST, select cell R11C4 and pull down the Edit menu. Choosing the Paste function makes a dialog box containing all of *Multiplan*'s built-in functions appear. Move the scroll box in the dialog box until SUM() appears. Select SUM, and it will be pasted into the formula bar.

Pull down the Edit menu again. This time choose the Paste Name option. A dialog box containing all assigned names appears. So far you've only assigned one name, COST. Choose COST, and it will be pasted into SUM() to form the function SUM(COST). Finally, press Enter to enter the completed function in cell R11C4.

Name the cells in the Today's Value column **VALUE** and those in the Dollar Change column **CHANGE**. Then define the SUM functions for each of these columns. You can use the Paste Name and Paste Function options, or you can type in the functions and names directly as = SUM(VALUE) and = SUM(CHANGE). When you are finished, the worksheet will look like the one in Figure 3.

Filling In the Blanks

The table is now complete. Just enter the necessary information about the stock symbol, price per share, and number of shares into the appropriate cells in columns 1, 2, 3, and 5, and leave the rest to *Multiplan*.

Figure 4 shows the data for a completed sample worksheet, but if you enter this data in the appropriate cells in columns 1, 2, 3, and 5, you'll see a worksheet filled with crosshatches, as in Figure 5. This indicates that some columns are too narrow to accommodate the data to be entered in them.

You can save some space by formatting columns 4, 6, and 7 to display their information with no decimal places. (Select those columns and then pull down the Format menu and choose the Number of Decimals... option.)

The completed worksheet in Figure 4 almost fits on one screen. You will have to use the scroll box along the bottom edge of the worksheet to see the bar graphics in column 9.

Setting Column Widths


The final step is adjusting the column widths so that all the data is visible. (Regardless of what appears on the screen, the actual number will be in the Mac's memory, where it is used in all calculations.)


With the mouse, adjusting individual column widths is easy. Just move the pointer to the horizontal box containing the column numbers at the top of the screen. The pointer icon will change to indicate that you can use the pointer to adjust the column width. Position the pointer on the boundary of the column you want to change, press and hold down the mouse button, and drag the boundary in the desired direction. Expand each column until the number replaces the crosshatching. Expand the right-hand column containing the bar graphs to 30 or 40 characters.

Locking Up the Worksheet

Now that your worksheet is finished, you may want to go one step further and take out a little insurance. *Multiplan*'s Protect Document command protects your worksheets from accidental changes. This insurance is particularly important in an office in which the person who enters data to update a worksheet may be unfamiliar with the procedures involved.

Multiplan lets you specify a personal password for each worksheet. Once a password has been assigned, a protected worksheet can be viewed but not changed by anyone who doesn't know the password. If you forget your own password, you won't be able to unlock the worksheet either, so choose a password that is easy to remember. You might use something from the spreadsheet itself—for example, the first letters of the title of the spreadsheet (PVW in this example). If your worksheet doesn't require protection, you can accept the blank line in the dialog box, and a password won't be required to unprotect the worksheet.

Before you can protect the portfolio valuation worksheet with the Protect Document command, you have to unprotect those cells where data will be entered when you update the worksheet. Since not all of the cells are adjacent, you must use the  key in tandem with the mouse button.

Begin by positioning the pointer on cell R3C2, where the date will be entered, and clicking the mouse button. Cell R3C2 is now selected. Move the pointer to cell R7C5 in the Today's Price column. Hold down the  key and the mouse button simultaneously and drag the pointer down column 5 until you have selected all the data entry cells in that column.

Now pull down the Options menu and choose Remove Cell Protection. The selected cells are unprotected, and you are ready to protect the rest of the worksheet by choosing the Protect Document command from the Options menu.

When Protect Document is executed, the appearance of the *Multiplan* screen changes dramatically. Row numbers, column numbers, and the dashed lines that divided the screen into a grid of cells disappear. The unprotected cells are underlined, and if you don't click the mouse button, the data entry range appears in reverse video (see Figure 6).

Updating the Worksheet

To update a protected worksheet, all you have to do is begin typing. The active cell is the date cell, the first cell you specified in the unprotected range. Type the date and press Enter. The active cell automatically moves to the next unprotected cell, the first cell under Today's Price. *Multiplan* is now ready for you to update the prices. Each time you finish an entry, press Enter; the active cell moves to the next cell, ready to receive your next entry.

If you want to skip a cell, press Enter without making an entry. The active cell will move on. If you want to go back to a previous cell, use the Shift-Enter combination.

The Sky's the Limit

The simple worksheet just described is only a beginning. With the ability to unprotect cells randomly on the worksheet and move the active cell through these cells in the order they were selected, you can create easy-to-use data entry forms. You can set up invoices, budgets, profit and loss statements, cash flow statements, and all types of report forms.

Just remember to unprotect the data entry cells before protecting the worksheet. Then anyone can update the worksheet without the risk of unauthorized or unintended changes.

Multiplan
Microsoft Corporation
10700 Northup Way
Bellevue, WA 98004
206/828-8080
List price: \$195

Congratulations Apple Computer



dilithium Press

P.O. Box 606
Beaverton, OR 97075
(800) 547-1842

Dilithium Press, the #1 publisher of easy-to-read computer books, will release books and book/software packages for the Macintosh in Spring, 1984. Watch for our announcements in *Macworld* and check your local bookstore or computer store for our new titles.

Tips for Mac Writers

A collection of 25 practical techniques for using MacWrite

Andrew Fluegelman

Although most word processing programs have similar functions and features, each has different ways of implementing them. *MacWrite* is an easy-to-use program, but there are some ways to improve upon its features, as you will find out when you experiment with it. The following tips will help you use *MacWrite*'s unique editing and formatting features more efficiently.

Deleting

Whatever your writing style, you will probably use the delete function more often than any other. There are several ways to make deletions with *MacWrite*, and each is particularly useful in different writing and editing situations.

- While you are keying in text, use the Backspace key to correct mistyped characters or to remove a word or two. This technique lets you keep your hands on the keyboard so that your writing flow is unbroken. Most writers probably use this technique naturally, but if you're new to word processing or keyboard writing, you might be tempted to use some of the more powerful editing functions when a simple destructive backspace is sufficient.

- A common editing function is deleting stray characters from a body of text. In this situation, using the Backspace key is the quickest and most accurate way to delete. Position the pointer so that the insertion point is to the right of the character(s) to be deleted, click the mouse button, and press Backspace. This method is easier than using the mouse to select the characters to be deleted because you only have to set the position of the pointer once. Selecting characters with the mouse involves

dragging the pointer from the starting point to the ending point of the selection.

- When you want to delete a larger portion of text such as several words, you should select the phrase to be deleted by dragging the pointer over that text. If you have no use for the text you're deleting, pressing the Backspace key is the fastest way to get rid of it. Keep in mind that this operation will not put the deleted text into the Clipboard, so you can't reuse it; however, choosing the Undo command from the Edit menu allows you to recover deleted text immediately (pressing ⌘-Z also undoes the previous action).

- If you are deleting and substituting new text, take advantage of the powerful *MacWrite* feature that automatically de-

If you're deleting a substantial portion of text, don't cut it without pasting it somewhere.

letes selected text and inserts new text in its place. In other words, if you type "The quick green fox..." you can make your correction by selecting the word *green*, and then typing *brown*. Overall, this innovative feature is the most useful *MacWrite* editing technique.

- You can also select a word by double-clicking the mouse button on it. If you've badly mangled a word such as *subsitutin*, for example, don't go to great

lengths to salvage the correct characters; select the whole word by double-clicking and retype it. This tip also applies to short phrases you want to repair or change, such as "The brown quick foxes." You can select and retype "quick brown fox" in less time than it would take you to analyze how to go about fixing the original.

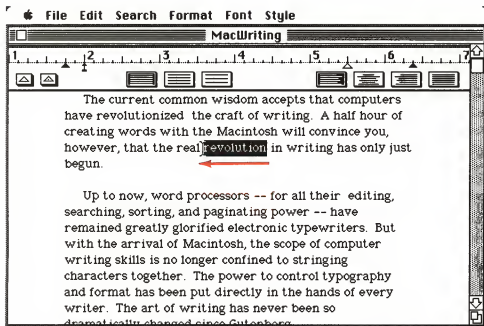
- None of the above three deletion techniques will cut deleted text to the Clipboard, so you have to be careful that you don't delete anything of value. You can use these techniques to your advantage, however, if you want to delete text and already have some in the Clipboard that you're saving for a later paste action.

- You might find it more convenient to select phrases for deletion by dragging the pointer backward, from the end point of the selection to the beginning. Since you read from left to right to decide what to select, your eye is already grounded at the end of the selection. It's easier to start dragging at that point and work backward than to find the beginning point again and still have to keep an eye out for the end point. Of course, this technique is a matter of personal taste, and might depend on whether you're right-handed or have perfected your tennis backhand stroke, but give it a try (see Figure 1).

- You might be surprised at such an exhaustive analysis of how to select and delete text. It's worth your time, however, to become familiar with the techniques that will serve you best while you are writing and to make them automatic reflexes. The tennis analogy is really not so farfetched. If you practice your strokes at the beginning,

Figure 1

You can select text by dragging the pointer backward, from the end point to the beginning point of the selection



you'll be a much better player in the word processing game.

- This last deletion tip has nothing to do with mechanical efficiency: If you're deleting a substantial portion of text—a paragraph or even a few sentences—don't cut it without pasting it somewhere, either to the end of your document or to the Scrapbook. Those phrases and ideas may be useful somewhere else in your text or perhaps in some other writing project. When you're ready for your final edit, you can save the unused fragments to separate files if they're still of value.

Editing

When you get to moving text around in a MacWrite document, as opposed to making spot corrections, some other techniques come into play.

- The best way to select a whole paragraph is to drag the pointer down the left margin. This procedure selects an entire line at a time and ensures that you'll include the invisible but crucial Return character at the end of each paragraph.

- If you're selecting a large portion of text, make use of the Shift key to extend your selection. As explained in the MacWrite manual, you first make a selection in the normal way, then move to another portion of your document, hold down the Shift key, and click the mouse button. Every-

thing from your original selection to the position of the pointer will be selected. You can use the scroll box to get to the end point of your selection. This technique is best for selecting everything from a point in the middle of a document to the end or the beginning.

- Keep in mind that double-clicking is an accurate way to select whole words and that it works with extended selections as well. Double-click to start your selection on a word boundary. Then move to the end of the area to be selected and Shift-double-click.

- By all means, develop the habit of using the ⌘-x, ⌘-c, and ⌘-v key combinations while you are editing. The Edit menu lists these special key combinations next to the corresponding command. Once a portion of text is selected, holding down the ⌘ key and the x key simultaneously cuts the selection to the Clipboard. ⌘-c copies the selection to the Clipboard, and ⌘-v pastes whatever is in the Clipboard at the current pointer position. The advantage of using these ⌘-key combinations (rather than choosing Cut, Copy, or Paste from the Edit menu) is that you can keep the pointer positioned within the text area. If you use these shortcuts for cutting and pasting, your editing efficiency will triple.

- Remember the Undo command.

This powerful MacWrite feature lets you reverse cuts and pastes. Undo will also take back the characters typed since the previous program command. This capability can be useful if you start typing (and there-

Chart of Typefaces

MacWrite provides you with almost two thousand possible combinations of type fonts, sizes, and styles. Although you'll be using only a fraction of those combinations, you'll still have to exercise more than

a little discretion in making your typographic selections. The following chart reproduces printed output from a *MacWrite* document, showing some of the most legible and attractive combinations.

New York is a serif font suitable for text. It prints best in:

9 point - ABCDEFGabcdefg12345 **ABCDEFGabcdefg12345** *ABCDEFGabcdefg12345*

12 point - ABCDEFGabcdefg12345 **ABCDEFGabcdefg12345**
ABCDEFGabcdefg12345 **ABCDEFGabcdefg12345**

14 point - ABCDEFGabcdefg12345 **ABCDEFGabcdefg12345**
ABCDEFGabcdefg12345 **ABCDEFGabcdefg12345**

18 point - ABCDEFGabcdefg12345 **ABCDEFGabcdefg12345**
ABCDEFGabcdefg12345 **ABCDEFGabcdefg12345**

24 point - ABCDEFGabcdefg12345 **ABCDEFGabcdefg12345**
ABCDEFGabcdefg12345 **ABCDEFGabcdefg12345**

18 point - ABCDEabcde1234 **ABCDEabcde1234**
ABCDEabcde1234 **ABCDEabcde1234**

24 point - ABCabc123 **ABCabc123**
ABCabc123 **ABCabc123**

Geneva is a sans-serif font suitable for text. It prints best in.

9 point - ABCDEFGabcdefg12345 **ABCDEFGabcdefg12345** *ABCDEFGabcdefg12345*

12 point - ABCDEFGabcdefg12345 **ABCDEFGabcdefg12345**
ABCDEFGabcdefg12345 **ABCDEFGabcdefg12345**

14 point - ABCDEFGabcdefg12345 **ABCDEFGabcdefg12345**
ABCDEFGabcdefg12345 **ABCDEFGabcdefg12345**

18 point - ABCDEFGabcdefg12345 **ABCDEFGabcdefg12345**
ABCDEFGabcdefg12345 **ABCDEFGabcdefg12345**

24 point - ABCDEabcde1234 **ABCDEabcde1234**
ABCDEabcde1234 **ABCDEabcde1234**

24 point - ABCabc123 **ABCabc123**
ABCabc123 **ABCabc123**

Toronto is an expanded serif font. It prints best in:

12 point - ABCDEFGabcdefg12345 **ABCDEFGabcdefg12345**
ABCDEFGabcdefg12345 **ABCDEFGabcdefg12345**
14 point - ABCDEabcdefg12345 ABCDEabcdefg12345
ABCDEabcdefg12345 **ABCDEabcdefg12345**
18 point - ABCDEabcde1234 ABCDEabcde1234
ABCDEabcde1234 **ABCDEabcde1234**
24 point - ABCabc123 ABCabc123
ABCabc123 **ABCabc123**

Monaco is a monofont (all letters same width). It prints best in:

9 point - ABCDEFGabcdefg12345 **ABCDEFGabcdefg12345** *ABCDEFGabcdefg12345*
12 point - ABCDEFGabcdefg12345 **ABCDEFGabcdefg12345**
ABCDEFGabcdefg12345 **ABCDEFGabcdefg12345**

Venice is a cursive font. It prints best in:

12 point - *ABCDEFGabcdefg12345* **ABCDEFGabcdefg12345**
ABCDEFabcdefg12345 **ABCDEFabcdefg12345**

Chicago is the Mac's system font. It prints best in:

12 point - ABCDEFGabcdefg12345 **ABCDEFGabcdefg12345**
ABCDEFabcdefg12345 **ABCDEFabcdefg12345**

Athens is a display font. It prints best in:

18 point - ABCDEabcde1234 ABCDEabcde1234
ABCDEabcde1234 **ABCDEabcde1234**

London is an Old English font. It prints best in:

18 point - **ABCDCabcde1234** **ABCDCabcde1234**
ABCDCabcde1234 **ABCDCabcde1234**

fore inserting.) in the middle of a paragraph and want to restore the paragraph to its original state. Undo can be invoked from the keyboard using **⌘-Z**.

- Remember that there are four places to which you can move and store text while you are editing: the Clipboard, the Scrapbook, the beginning of your document, and the end of your document. Each location has specific uses. The Clipboard is volatile—its contents are replaced with each successive Cut or Copy. The Scrapbook stores text (or pictures) permanently and is convenient for inserting stock text selections several times within a document. Don't overlook the beginning and end of your file as convenient holding areas for text that you know you will be working into your document at some point.

- If you are doing a good deal of cutting and pasting, you might want to set up your screen with both the text area and the Clipboard in full view, serving as a permanent on-screen reference (see Figure 2).

- The same screen arrangement is helpful for aligning the page, date, and time icons properly in the header or footer windows. With both the header and footer windows in view, you can adjust the icons and then click the text window to see the effects immediately.

Searching

The MacWrite Find and Change functions are useful if you need to change the name of a character throughout your short story, but some of the best uses for the Search capability don't involve changes per se.

- If your text includes a key word that is long or difficult to type, substitute a distinctive, easily typed symbol while you're doing your initial writing. Then change it during your final edit. For example, you could use the @ symbol and later change it to *Tchaikovsky*.

- MacWrite does not feature place markers, but you can make use of the Search function to guide you through your

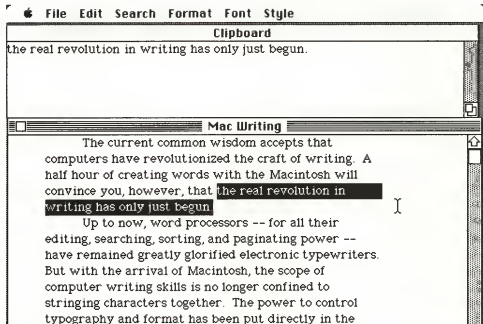


Figure 2

Having the Clipboard in full view provides an on-screen reference during cut-and-paste operations

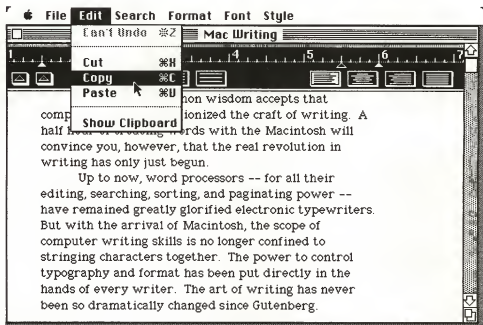


Figure 3

Copying a ruler with a specified format into the Clipboard for insertion elsewhere in the document

document. If you are moving a selected portion of text to the beginning or end of your document, embed a special symbol, such as the ++ symbol, in the text before repositioning the pointer to complete the paste operation. Then you can find your original position easily by searching for the ++ symbol.

- When you want to replace a word globally throughout your document with the Change All command, be careful that you don't accidentally replace fragments of longer words. Suppose, for example, that you decided to change all occurrences of *Mac* to *Macintosh*. If the word *Macintosh* were already in your document, it would come out as *Macintoshintosh*. To avoid this situation, you could include spaces as part of the Find string: *Mac[space]*. Then your search would look only for whole words. However, your search would not find occurrences of *Mac* at the end of a sentence.

Another strategy would be to change all instances of *Macintosh* to *Mac* first, and then change all the *Mac*'s back. You would still have to search for *Mac[space]*, however.

- After wading through the above scenarios, you should see clearly that global changes can have unexpected results. It's usually wise to take the time to confirm each modification with *MacWrite*'s Change, then Find option.

Formatting

MacWrite has powerful formatting capabilities that let you set margins, indentation, tabs, line spacing, and type styles. You can streamline your formatting chores by using some of the following techniques.

- Most writers have standard writing applications—letters, memos, draft manuscripts—that require different formats. Take the time to determine the best format for each of these situations, then save the formats in separate files and use one of those files to start each type of writing project.

For example, suppose you like to prepare drafts with 2-inch margins on both sides of the page, double-spaced, using 14-point Athens type. Start *MacWrite* and open a new document. You'll see a blank document called *Untitled* in the *MacWrite* window. Without typing any text, move the left margin marker to 2 inches and the right to 6½ inches, click on the double-

space icon, and select 14-point Athens type. Then choose Save from the File menu and name the document Draft Format. Now every time you want to begin a draft, start *MacWrite* by double-clicking on the icon labeled Draft Format. You'll automatically create a document with your chosen formatting features.

When you are ready to save your document, don't choose the Save command. Instead, choose Save As... from the File



*The Scrapbook
stores text (or pic-
tures) permanently
and is convenient for
inserting stock text
selections.*

menu. You'll be asked to give a name under which to save the current document. At this point you should give your written document a distinctive name, such as Draft Story 23.

Once you've copied and saved the document under a new name, choose Quit from the File menu, making sure that you don't save your original Draft Format document with all the text you added. The next time you want to write a draft, start with the Draft Format document again; it will once again present a clean slate, formatted according to your specifications.

- You can expand the above technique to create a correspondence format file that produces your own customized letterhead, complete with a *MacPaint* graphic, your address, and the date.

- If you are creating a document that uses several formats, such as an outline or a screenplay, you'll want to be able to switch formats quickly and easily. Before you start writing, make up the different rulers that you'll need and place them in a "ruler well" at the beginning or end of your working file. When you need a certain

format, such as indented single-spaced, you can go to the well, copy the appropriate ruler to the Clipboard, and paste it in place (see Figure 3). This technique is much faster than inserting new rulers and adjusting the margins each time you need them. (To select a ruler, click on the portion of the ruler with the inch markings.)

- Note that the rulers carry formatting information, but not type style and size specifications. If you want to make those elements part of your multiple formats, you should include at least one blank line (two Returns) with each ruler before you copy it to the Clipboard. Then when you paste in the ruler you'll include a spot in your document that carries the appropriate type specifications. As you insert text at that spot, it will assume the specified format. This technique will work, but it can be difficult at first to know exactly where the boundaries are between different type fonts and styles. Experiment and practice.

Bear in mind that the *MacWrite* window is not as wide as a standard sheet of paper. In relation to an 8½-inch-wide sheet of paper, the *MacWrite* ruler starts at 1 inch and ends at about 7¼ inches. Thus, there are margins beyond what you see on the screen. If you adjust the screen to show pleasing margins, your printed text may be too narrow. (Don't be afraid to format your documents with adequate "white space," however.)

- You should also note that *MacWrite*'s default format (the way the program starts out, without any user specification) does not include any headers or footers. Unless you specify otherwise, your printed document will start at the top of each page and continue to the bottom. If you want top and bottom margins, call up the Header and Footer windows and insert two or three Returns in each. These specifications can be included in your document format files, as noted above.

SimpleSketch

*Programming the Mac in
BASIC and Pascal*

Lon Poole

SimpleSketch is a short program that demonstrates the kind of control the Macintosh User Interface Toolbox gives you over the Mac, whether you program in Pascal or BASIC. It lets you construct rectangles of any size in the program's output window. You use the mouse to specify the top-right and bottom-left corners of the rectangle. One version of the program is written in Macintosh Pascal (MacPascal), another in Microsoft BASIC (MBASIC), and a third in Macintosh BASIC (MacBASIC). After exploring each version of the program, you'll learn how to modify the original program to draw ovals and circles in a way that *MacPaint* doesn't duplicate.

Using the Program

Drawing rectangles with SimpleSketch is the same as drawing them with *MacPaint*. First, you move the pointer to the spot where you want to place the top-left corner of the rectangle and press the mouse button. The program remembers those coordinates. Then you move the pointer to the spot where you want to place the bottom-right corner of the rectangle and release the mouse button. Figure 1 shows the type of objects SimpleSketch can draw.

As long as you hold down the mouse button, a flickering outline shows the current size and shape of the rectangle. To produce the flickering effect, the program draws a rectangle in black and then quickly erases the image by redrawing the rectangle in white. The program keeps drawing and erasing the rectangle as long as you hold down the mouse button. Moving the mouse while you hold down its button modifies a rectangle's proportions by

changing the location of its bottom-right corner. The program recognizes those changes and resizes the flickering outline accordingly. When you release the mouse button, the program redraws the rectangle permanently in black, using the proportions of the last flickering outline.

MacPascal SimpleSketch

Listing 1 shows the MacPascal version of the SimpleSketch program. Notice that the command words are in boldface and

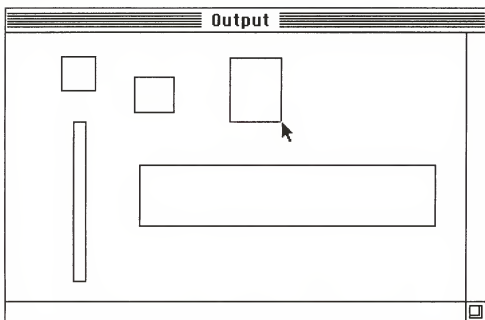
The SimpleSketch program uses the frameRect procedure from the QuickDraw Toolbox unit to draw rectangles and squares.

the program lines are indented. MacPascal does this automatically to make the listing easier to read. Capitalization is optional, except in the boldface words. MacPascal considers an uppercase letter and its lowercase counterpart to be the same character (unless they appear between quotation marks). Thus, you can use capital letters to improve program readability. MacPascal, like other versions of Pascal,

does not require line numbers. (Note: the numbers to the left of the program lines in the listing pertain only to the following description of the program.)

1. Declares that four variables will have integer values. Variables *left* and *top* will keep track of the coordinates for the point at the top-left corner of the rectangle. Variables *right* and *bottom* will monitor the coordinates of the bottom-right corner.
2. Sets the size of the pen point that will draw the rectangles. Procedure *penSize* is predefined; its first parameter sets the pen-point width and its second parameter the pen-point height.
3. Starts the program loop that repeats the program endlessly so that you can draw as many rectangles as you like. (To halt the program, choose the Halt command or click the displayed Halt button.)
4. Sets a pen mode that enables a flickering outline to be drawn. The predefined *penMode* procedure is a standard element of the QuickDraw Toolbox unit. Its single parameter defines how an object to be drawn will interact with the existing images in the window. The *patXor* option lets a program erase an object without disturbing other images by displaying the object on top of itself.
5. Starts a loop that waits for you to press the mouse button. Before you press the mouse button, another predefined procedure, *getMouse*, monitors the mouse location and assigns the coordinates to variables *left* and *top*. When you press the mouse button, the loop ends, fixing the top-left corner of the rectangle.

Figure 1
Sample output from
SimpleSketch.



6. Starts a loop that repeats as long as you hold down the mouse button.

7. Assigns the latest coordinates for the bottom-right corner of the rectangle to variables *right* and *bottom*.

8. Uses the predefined procedure *frameRect* to draw the rectangle defined by the current corner coordinates.

9. Draws the same rectangle a second time at the same location. Because the drawing mode at this time is *patXor*, the second drawing erases the first, without disturbing the rest of the screen. As a result, you see the rectangle flash briefly on the screen.

10. When you finally release the mouse button, the program changes the drawing mode, again with the *penMode* procedure. This time it uses the *patCopy* mode, which always makes black lines black, regardless of what they are drawn over.

11. This time the *frameRect* procedure draws the rectangle permanently at its final location.

12. The *until* statement marks the boundary of the loop started by the *repeat* statement. The *end* statement marks the end of the program.

Microsoft BASIC SimpleSketch

The output of the MBASIC version of SimpleSketch will look the same to its user as the MacPascal version (see Listing 2). Notice that all the program lines have line numbers. Also, this listing uses capital letters everywhere except in program remarks. You can type the program into the Mac using any combination of upper- and lowercase letters, but MBASIC will convert lowercase letters to capitals in the listing

(unless they are part of a program remark or enclosed by quotation marks).

Here's how the MBASIC program works, line by line:

20. Declares that all variables starting with the letters *L*, *T*, *R*, and *B* will have integer values. Alternatively, this statement could be omitted and the name of each integer variable suffixed with a % character.

30. Clears the output window.

40. Sets the size of the pen point. The first parameter of *PENSIZE* defines the width; the second, the height.

50. Sets the pen mode. Mode 10, as it is called in MBASIC, works the same as the *patXOR* mode in MacPascal. In this mode, drawing the same object twice in the same place erases it without erasing other images.

60-70. The *MOUSE* function has different effects depending on the value of the parameter passed to it. *MOUSE(1)* and *MOUSE(2)* determine the mouse's current coordinates, and variables *LEFT* and *TOP* keep track of them.

80. Until you press the mouse button, the program loops between lines 60 and 80. *MOUSE(0)* checks the mouse button status and has a value of 0 if the button has not been clicked since the last time *MOUSE(0)* was executed.

90-100. When you press the mouse button to select the top-left corner of the rectangle you want to draw, the program goes into another loop. First, it gets the mouse coordinates and assigns them to variables *RIGHT* and *BOTTOM*.

110. MBASIC expects the rectangle's corner coordinates in the first four elements of an integer array; this program uses array *R*().

120. Calls Toolbox procedure *frameRect* to draw the rectangle whose coordinates are in array *R*().

130. The second call to *frameRect* erases the rectangle just displayed; the result is a flashing rectangle.

140. As long as you hold down the mouse button, the program loops between lines 90 and 140, flashing rectangles sized by the current mouse coordinates. *MOUSE(0)* has a value of -1 as long as the mouse button is held down.

150-160. Changes the drawing mode and permanently draws the rectangle at its final location.

170. Branches back to wait until you select the top-left corner of another rectangle. (To halt the program, press the **⌘** and **c** keys simultaneously.)

MacBASIC SimpleSketch

From a program user's standpoint, the MacBASIC version of SimpleSketch works just like the MacPascal and MBASIC versions (see Listing 3). But a comparison of the different versions of this program reveals disparities even between the two BASIC versions. Unlike MBASIC, MacBASIC does not require line numbers. (Note: the numbers to the left of the program lines in the listing pertain only to the following description of the program.) Also, MacBASIC displays listings in upper- and lowercase letters, although it treats capital letters and their lowercase counterparts as the same characters (unless they are between quotation marks). MacBASIC, unlike MacPascal, does not automatically format listings. To indent program lines for better readability, you have to type the spaces yourself by pressing the space bar.

program SimpleSketch

```

var
1  top, left, bottom, right : integer;
begin
2  penSize(1, 1);
3  repeat
4  penMode(patHor);
5  while button = false do
      getMouse(left,top);
6  while button do
      begin
7  getMouse(right,bottom);
8  frameRect(top, left, bottom, right);
9  frameRect(top, left, bottom, right);
      end;
10 penMode(patCopy);
11 frameRect(top, left, bottom, right);
12 until false;
end.

```

Listing 1

MacPascal version of SimpleSketch. Command words are in boldface and program lines are indented.

```

10 REM SimpleSketch Program
20 DEFINT L,T,R,B
30 CLS
40 CALL PENSIZE(1,1)
50 CALL PENMODE(10)
60 LEFT=MOUSE(1)
70 TOP=MOUSE(2)
80 IF MOUSE(0)=0 THEN GOTO 60
90 RIGHT=MOUSE(1)
100 BOTTOM=MOUSE(2)
110 R(1)=LEFT:R(0)=TOP:R(3)=RIGHT:R(2)=BOTTOM
120 CALL FRAMERECT(URRPTR(R(0)))
130 CALL FRAMERECT(URRPTR(R(0)))
140 IF MOUSE(0)=-1 THEN GOTO 90
150 CALL PENMODE(8)
160 CALL FRAMERECT(URRPTR(R(0)))
170 GOTO 50

```

Listing 2

Microsoft BASIC version of SimpleSketch. MBASIC uses capital letters everywhere except in program remarks.

Notice that the MacBASIC program does not declare any variable types. It determines a variable's type from the last character of its name. All of the variables in this program have a % character suffix, which identifies them as integer types.

The MacBASIC program works as follows:

1. Sets the pen-point size. The *set pensize* statement has two arguments: the first sets the pen-point width, and the second the height.

2. Starts the main program loop. The program goes through the loop once each time you draw a square.

3. Sets the pen mode that lets the program stretch and shrink the rectangle as it follows the mouse movement.

4. Waits for you to press the mouse button, thereby fixing the top-left corner of the rectangle you want to draw.

5. Gets the coordinates of the top-left corner from functions *mouseX* and *mouseY* and assigns them to variables *left%* and *top%*.

6. This line has a label, *Drawagain*, that a later statement branches to as long as you hold down the mouse button.

7. Assigns the coordinates of the current bottom-right corner to variables *bottom%* and *right%*.

8. Flashes the rectangle defined by the currently selected corner points. The first *frameRect* statement draws the rectangle, and the second erases it.

9. Repeats the rectangle-flashing loop as long as you hold down the mouse button.

10. The program changes the pen mode to 8 when you release the mouse button. This mode, which is the same as MacPascal's *patCopy* mode, always draws black lines in black. This time the *frameRect* procedure draws the rectangle permanently at its final location.

```

1 — set pensize 1,1
2 — do
3 — set penmode 10
4 — btnwait
5 — left%=mouseX
   top%=mouseY
6 — Drawagain: right%=mouseX
7 — bottom%=mouseY
8 — frameRect left%,top%;right%,bottom%
   frameRect left%,top%;right%,bottom%
9 — if mouseb then goto Drawagain:
10 — set penmode 8
    frameRect left%,top%;right%,bottom%
11 — loop

```

Listing 3

MacBASIC version of SimpleSketch. Unlike MBASIC, MacBASIC does not require line numbers.

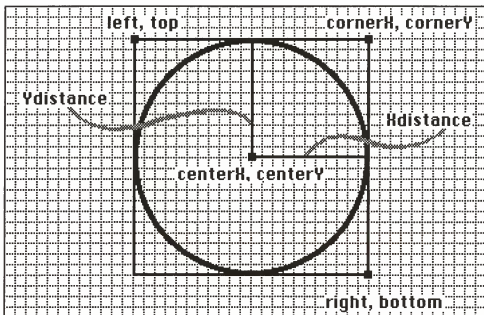


Figure 2

OvalSketch determines the top, left, bottom, and right coordinates given the center point and any corner point of the rectangle into which an oval fits.

11. The program branches back to the earlier *do* statement so that you can draw another rectangle. (To halt the program, choose the Halt command.)

The Oval Alternative

Like *MacPaint*, the original SimpleSketch program uses the *frameRect* procedure from the QuickDraw Toolbox unit to draw rectangles and squares. With *MacPaint*, you can also draw an oval by selecting the rectangle it fits in, first choosing the upper-left corner and then the bottom-right corner. The key to drawing ovals and circles is another QuickDraw procedure, *frameOval*.

Drawing ovals the same way *MacPaint* does is easy in the MacPascal program; simply change the procedure name in each of the three program lines in which it occurs. With a little more work, you can enhance the original SimpleSketch program so it does something that can't be done in *MacPaint*. Listing 4 shows a MacPascal program, *OvalSketch*, that lets you draw an oval by selecting the center point and any corner of the rectangle into which it fits. Figure 2 shows how it works, and Figure 3 illustrates the program output.

The following description explains how *OvalSketch* works, with emphasis on the modifications:

1. This program needs some additional variables to keep track of the center point, the current-corner point, and the horizontal and vertical distances between the center and corner points.

2. Notice that this program will draw thicker lines than the original.

3. Starts a loop that lets you draw ovals indefinitely.

4. Sets the pen mode for flashing interim ovals.

5. Waits for you to select the center point by pressing the mouse button.

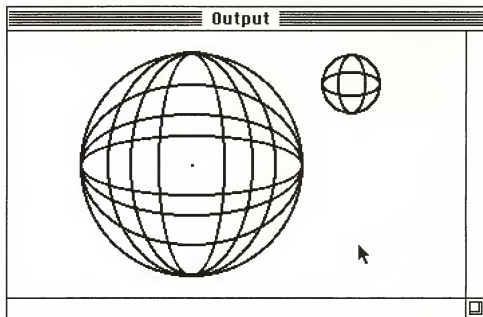


Figure 3
Sample output from *OvalSketch*.

6. Starts the loop that repeats as long as you hold down the mouse button.

7. Assigns the coordinates of the current-corner point to variables *cornerX* and *cornerY*.

8. Computes the distance from center to corner. Without the *ABS* function, distances are negative when the corner is above or to the left of the center point.

9. Computes the coordinates for the corners of the rectangle in which to draw the selected oval or circle. The top-left corner is above and to the left of the center point, and the bottom-right corner is below and to the right of center.

10. Flashes the oval defined by the current center and corner points.

11. Changes the pen mode and draws the oval or circle in its final proportions.

12. Allows you to draw another oval. (To halt the program, choose the Halt command or click the displayed Halt button.)

Further Experimentation

The simplicity of these programs may fool you into thinking that programming an application like *MacPaint* wouldn't be so tough after all. To keep things in perspective, try modifying SimpleSketch or *OvalSketch* to do the following:

- Use thicker or thinner lines.
- Draw rectangles above or to the left of the starting point (SimpleSketch).
- Draw circles by selecting a center point and any point on the circumference. The pointer must stay on the circumference as you drag it away from the center. (Hint: use the *sqr* function and the fact that the length of the hypotenuse of a right triangle is equal to the square root of the sum of the squares of its sides.)

• Draw a variety of other shapes using the QuickDraw procedures *frameRoundRect*, *paintRoundRect*, *paintOval*, and *paintRect* in place of *frameRect*.

• Draw straight lines. Use the QuickDraw procedure *moveTo(x,y)* to anchor a starting point and *lineTo(x,y)* to draw a line from there to another point.

• Allow freehand drawing.

• Display a palette of drawing options.

Allow the program user to use the mouse to select the line thickness and the shape of the object to draw.

These experiments only begin to explore the ways you can enhance the basic SimpleSketch program. As you work out ways to effect these improvements, new ideas will probably come to mind. Were you to continue adding features to the program, you would end up with thousands of program lines distributed among dozens of independent procedures that would collectively provide sketching and drawing capabilities similar to those in *MacPaint*.

Macintosh Pascal
Think Technologies
Liberty Square
Danvers, MA 01923
617/777-5226
Distributed by Apple Computer, Inc.
List price: \$99

Macintosh BASIC
Apple Computer, Inc.
20525 Mariani Ave.
Cupertino, CA 95014
800/538-9696; in California
800/662-9238
List price: \$99

Microsoft BASIC
Microsoft Corporation
10700 Northup Way
Bellevue, WA 98004
206/828-8080
List price: \$150

```
program OvalSketch;
var
1  centerH, centerY, cornerH, cornerY : integer;
   Hdistance, Ydistance : integer;
   left, top, right, bottom : integer;
begin
2  penSize(2,2);
3  repeat
4  penMode(patHor);
5  while button=false do
      getMouse(centerH, centerY);
6  while button do
      begin
7      getMouse(cornerH, cornerY);
8      Hdistance := abs(centerH - cornerH);
      Ydistance := abs(centerY - cornerY);
9      left := centerH - Hdistance;
      top := centerY - Ydistance;
      right := centerH + Hdistance;
      bottom := centerY + Ydistance;
10     frameOval(top, left, bottom, right);
      end;
11     penMode(patCopy);
      frameOval(top, left, bottom, right);
12 until false;
end.
```

Listing 4

OvalSketch, a MacPascal program that lets you draw an oval by selecting the center point and any corner of a rectangle into which it fits.

The 64K Treasure Chest

Lon Poole

Apple has gone to unprecedented lengths to attract software developers to the Mac and to persuade them to adopt the Mac desktop environment as the standard user interface. The benefits of this system are that developers can create better programs with less effort and that users will have a familiar working environment for all their Mac applications.

The usefulness of any computer depends on a wide variety of quality software. Apple has done several things to encourage people to develop application programs for the Macintosh, including the following:

- Developing guidelines for a consistent user interface.
- Supplying programs that implement the user guidelines.
- Using an operating system that gives application programmers extraordinary control over system events.
- Designing a unique way to greatly simplify the editing of dialog and menu wording.
- Making available a speedy and powerful development system based on the Lisa.
- Providing inexpensive expert advice and support to registered developers.

User Interface Guidelines

Designers of interactive application programs spend a great deal of time worrying about the user interface—the way the program interacts with its users. For the Mac, Apple has done the research and development, made the decisions, and come up with a flexible interface suitable for almost all applications. The Mac user interface, based on the desktop environment, includes icons, windows, pull-down menus, mouse pointing, displayed control buttons, dialog and alert boxes, basic text editing, and cut-and-paste editing (see “A Tour of the Mac Desktop”). The Macintosh user interface guidelines manual defines

the standard desktop environment. By following the advice in the manual, you can get a consistent user interface that people will understand and accept. Adopting the Mac user interface guidelines allows you to focus on your strong point—the application itself.

The User Interface Toolbox

Adopting the Mac user interface is one thing; implementing it is another. Imagine writing the programs that manage the Mac's desktop environment. Few application programmers would bother to work with the Mac user interface if they had to write those programs themselves.

Apple realized that simple truth, wrote the programs to manage the desktop environment, put them in a 64K ROM, and called the set of codes the User Interface Toolbox. This 64K treasure chest means you'll never have to design or code a user interface for the Mac.

Here's how the Toolbox works within an application. The application program uses one part of the Toolbox to poll system events such as mouse and keyboard input. When the program user presses a key, clicks the mouse, or inserts a disk, the application uses another part of the Toolbox to make the Mac react appropriately. The application sends a message that describes the event to the appropriate Toolbox procedure, which then effects a change on the electronic desktop. Of course, an application does not have to use the Toolbox to react; it can always use its own procedures.

Actually, Apple came up with more procedures to put in the Toolbox than would fit in the 64K ROM, so some of the Toolbox units will take up part of RAM if you choose to use them (see “The Toolbox” for a brief description of the Toolbox units).

Programming Languages

You can write some application programs in Macintosh Pascal (MacPascal) or Macintosh BASIC (MacBASIC). Both are interpreted languages, which provide an ideal debugging and fine-tuning environment. You can change your program and immediately run it to see the results of the changes.

There are some trade-offs in using MacBASIC or MacPascal. For one, interpreted languages are slower than compiled languages; however, both MacBASIC and MacPascal are fast enough for many applications. Also, the first versions of MacPascal and MacBASIC cannot turn over full control of the screen to an application pro-

gram. One solution uses two Macs; you write and edit a MacPascal program on one Mac and run it on an interconnected second Mac. The interpreter's menus and windows appear on the first Mac, and the application's menus, windows, and other graphics on the second. The other MacPascal development system uses one Mac. In this case, the MacPascal program takes over control of the menus and windows from the interpreter.

You can also develop Mac applications using Microsoft BASIC (MBASIC). Such applications will be subject to similar limitations as those written in MacBASIC or MacPascal, notably the inability to control the menu bar and all windows independently. MBASIC does afford a degree of compatibility with other computer systems that have versions of MBASIC. (See "Programming Preview" for a review of MacPascal, MacBASIC, and MBASIC.)

For now, most serious programmers will write Mac applications on a Lisa development system. This system consists of a Lisa with a hard disk, the Lisa Pascal compiler and linker, a procedure and function library for the Mac, a Mac on which to test the application program, and a cable to interconnect the Lisa and the Mac. (If you are developing more than one application, you may need more than one hard disk drive.) To work on an application, you write a Pascal program on the Lisa, compile it, and send it to the Mac for execution. While it runs on the Mac, you can look at the listing on the Lisa.



This 64K treasure chest means you'll never have to design or code a user interface for the Mac.

gram. Your program must share the menu bar with the interpreter and has limited control of the windows. Eventually, an enhanced version of MacPascal that features full Toolbox support will be available.

Within six months of the Mac's introduction date, you should be able to circumvent these problems. Apple currently has two plans on the drawing board that will make full application development using



A developer's kit will include all the tools you need to develop a Mac application program on a Lisa. Among the tools will be several complete, working sample application programs written in Lisa Pascal on 5¼-inch Lisa disks; Mac utility programs, including a resource file editor, text font editor, icon editor, and screen snapshot maker, all on Mac disks; a procedure and function library for Mac programs, on Lisa disks; and *Inside Macintosh*, a comprehensive technical manual containing user interface guidelines and Toolbox procedures. Some of these items may also be sold separately.

Apple plans to offer programming languages other than Pascal and BASIC. Within three months of the Mac's introduction an assembler and debugger will be available to those programmers who prefer coding directly in native 68000 assembly language. Aficionados of the C language can expect a compiler sometime within the first six months. The C compiler will make application program development feasible on a single Mac.

Developer Support

Apple has two programs that support developers. If you are actively developing software for the Mac, you can sign up for the Certified Developer Program. This program entitles you to a discount on Apple products, a subscription to a newsletter, invitations to technical seminars, and more. The Certified Developer Program itself is free, although you must meet certain criteria that establish your authenticity as a program developer. A fee will be charged for attending the seminars.

Certified developers can join the Registered Developer Program by paying a nominal yearly fee. Members of this program receive an electronic mail account that provides a bulletin board service and ongoing, high-level technical support from Apple product divisions.

Apple will also offer a course consisting of four or five in-depth classes to serious developers who use the Lisa development system. Enrollment will be restricted to advanced Pascal programmers and class size will be limited to around 20. Within six months after the introduction of the Mac, Apple will offer courses in Europe, Australia, and several major cities in the United States. Schedule announcements for the first course will go to regis-

The Toolbox

The User Interface Toolbox includes hundreds of procedures, which have been grouped into 15 units. Some units have just a few procedures, and others, dozens. The grouping is purely logical, not functional, and exists mainly to give application programmers an organized concept of the Toolbox. Programmers can freely choose any combination of Toolbox procedures from the various units and use them in any order. Naturally, not all combinations and sequences do something useful, and a few combinations will even crash the system. So along with the freedom and flexibility built into the Toolbox goes a degree of responsibility: programmers must be careful to avoid those procedural pitfalls.

Not all programming languages support every Toolbox unit, let alone every procedure. Lisa Pascal, in combination with a Macintosh procedure and function library, does support the full Toolbox. Assembly language and C programmers also have full access to the Toolbox. The first release of MacPascal, however, omits some of the rarely used, complicated, and potentially disastrous procedures, making the Toolbox easier to use and precluding inadvertent system crashes. For the same reasons, MacBASIC and MBASIC also limit Toolbox support.

The descriptions of the Toolbox units that follow will give you an idea of what an application program can do on the

Mac. The *User Interface Toolbox* manual has more complete information about the Toolbox units and the individual procedures in them.

- **Control Manager** displays buttons, check boxes, radio buttons, scroll bars, dials, and other application-specific controls. It can also determine a control's setting, change the setting, and indicate when someone uses the mouse to operate a control.

- **CoreEdit** handles sophisticated, paragraph-oriented text editing and formatting. It supports rulers, which determine margin justification and other parameters.

- **Desk Manager** supports the use of desk accessories from an application. Standard desk accessories include the Calculator, NotePad, Scrapbook, Clock, Key Caps, Puzzle, and Control Panel. You can also define your own accessories.

- **Dialog Manager** displays dialog and alert boxes and receives user responses to them.

- **Error Handler** provides a standard alert box for serious system errors that the application cannot handle.

- **Event Manager** reports mouse and keyboard activity, disk insertion, and other system events to application programs and other Toolbox units. It is also used for communications between Toolbox units.

- **Font Manager** primarily assists the QuickDraw unit in drawing text. Given the charac-

ters, font, style, size, scaling factor, and output device; the Font Manager supplies the pattern of dots that will draw the requested text.

- **Menu Manager** displays the menu bar. On command, it also displays a pull-down menu and takes the user's order.

- **QuickDraw** draws straight lines, rectangles, rounded-corner rectangles, circles, ovals, arcs, wedges, polygons, and arbitrary shapes, all either hollow or solid. With a single procedure call it can draw a complex picture made up of any combination of shapes and lines.

QuickDraw, with the aid of the Font Manager, also draws proportionally spaced text in any of several fonts. Characters can be any of several sizes (generally from 8 to 72 points) and can be in any combination of type styles including bold-face, italic, underlined, and outlined.

You can define multiple drawing ports and easily switch from one to another. Each port has its own coordinate system, cursor, character attributes, and screen location. QuickDraw provides complete image clipping at the edges of a drawing port.

You can also "draw" anything into an off-screen memory buffer, either for sending to an output device or for super high-speed display switching. It does all of these things fast enough to generate an interactive graphic user interface, animation, and complex text displays.

- **Resource Manager** accesses information in resource files that defines menus, fonts, icons, dialog boxes, alert boxes, or custom resources. Other Toolbox procedures use the Resource Manager to read and write resources. Consequently, many application programs never have to call the Resource Manager directly.

- **Scrap Manager** helps an application manipulate the Clipboard for cutting and pasting between applications, desk accessories, or an application and a desk accessory.

- **Segment Loader** swaps program segments between memory and the disk, allowing an application program to be larger than 32K, which is the maximum size of one segment.

- **TextEdit** supports basic text editing, including word wrap, selection, pointer movement, insertion, deletion, and replacement.

- **Toolbox Utilities** displays icons and drags icons, custom windows, or controls in response to mouse movement. Other utilities perform various functions such as fixed-point arithmetic, string manipulation, and logical operations on bits.

- **Window Manager** creates and manipulates windows on the screen. It handles overlapping windows and controls window activation/deactivation, closing, sizing, dragging, and scrolling.

tered Apple developers. Eventually, classes will be held approximately every three weeks at various locations around the world.

Development languages and systems will also be available from sources other than Apple. Microsoft, for example, has come up with its own development system, centered around the C programming language, that it uses for developing *Multiplan*, *Multibart*, *Multifile*, *Microsoft Word*, and other Microsoft applications. Microsoft will make its development system available to those few developers with whom it enters into a co-development arrangement.

Program Development

Writing application programs that make the most of the Mac requires a thorough understanding of the desktop environment; you'll have a lot to learn before you can even create a window and print your name in it. The user interface guidelines tell you what to do, and the Toolbox gives you the capability to do it, but neither one tells you how to do it. The trial-and-error approach is probably the easiest way to learn the "how-to" aspect. Start with a



For now, most serious programmers will write Mac applications on a Lisa development system.

working application, look at its code, see how and why it works, and use the same techniques to develop your own application.

The Lisa development system lets you do more than get ideas from working programs; you can also get working program code from them. Cut-and-paste editing makes this possible. You start with a working program in one window, cut out the parts unique to the application, and leave just the skeleton of the program. You can open more windows for other working programs, from which you can cut sections

of code to paste into the skeletal program. In another window you can write new procedures, and then cut and paste those routines into the skeletal program.

The Mac gives the application program much more control over external events than do most other systems. For example, you can control the keyboard autorepeat feature from an application program. No more coping with an operating system that blasts characters at you if you hold down a key. That's the type of thinking that pervades the whole system. Your application program controls system events, not vice versa.

You'll probably end up taking your application through several rounds of coding, refining the way it controls external events, windows, icons, menus, and other elements of the Mac user interface. This situation is especially true if you are

converting an existing program written for a computer with an old-fashioned character-based user interface and a rigid operating system. It should be fairly easy to get such a program running in its original style, using one window as it formerly used a whole screen and ignoring the menu bar. Then you can plug in elements of the Mac user interface one at a time, making your program easier to use.

Jeffrey Harbers, manager of the Mac applications development team at Microsoft, sums up program development for the Mac this way: "The Mac is like a Lamborghini, in that almost anybody can go out and sit in this car and look great. It takes very little work to look like a star. It's just a matter of opening the door and getting in, because they give you everything. But the difference is that this Lamborghini can do 160 miles per hour. It takes a little more skill and training to take it up to those speeds. People are going to find that the major limit is the limit of their imaginations."

It won't take long for software developers of all kinds, from large companies to individual programmers, to agree with Mr. Harbers. They'll accept the challenge and learn how to exploit the User Interface Toolbox to create some truly useful and usable applications. The Mac's electronic desktop may well engender the next breakthrough in personal computer applications.

For more information on software developer support, contact Apple Computer, Inc., 20525 Mariani Ave., Cupertino, CA 95014, 800/538-9696; in California 800/662-9238.

Invitation to Writers

..... *Macworld* invites writers to submit articles for publication. Articles should be focused to fit within the editorial structure of the magazine.

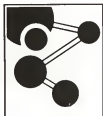
▲▲▲▲▲ *Macworld* consists of four editorial sections: *State of the Art*, *Review*, *Hands On*, and *Community*. *State of the Art* covers developing technologies for the Macintosh. In *Review*, new Macintosh hardware and software are critically and objectively analyzed by experienced users. *Hands On* offers how-to articles that instruct readers on Macintosh applications.

Community articles describe the impact of the Macintosh in workplaces, schools, and homes.

————— *Macworld* accepts commentaries for the guest editorial department, *NotePad*. User group members are invited to submit articles on any facet of their organization.

..... Articles may be submitted by query or as unsolicited manuscripts to Editorial Proposals, *Macworld*, 555 De Haro St., San Francisco, CA 94107. If you send a query letter, please include an outline and a short description of the article you are proposing, a brief account of your computer-related experience, and a sample of your work.

Data Communications for the Macintosh



Winterhalter, Incorporated

Welcome, Macintosh!
Since 1978, Winterhalter, Incorporated has been providing micro-to-main-frame communication solutions to the Fortune 1500 community. Now, Macintosh, it's your turn. Soon Winterhalter will be offering a complete line of data communication capabilities for the Macintosh. Look for more announcements soon.

Winterhalter, Inc.
P.O. Box 2180
Ann Arbor, MI 48106
800-321-7785

Open Window

An exchange of Macintosh discoveries

Edited by Andrew Fluegelman

Open Window is a forum for sharing discoveries about the Macintosh. We invite and encourage readers to make contributions to this department. Perhaps you've come up with a nifty routine, gained some insight into how the Mac or an application program works, or even written a short program that performs a useful function or creates an interesting diversion. Tell us about it and we'll pass your discovery along to other members of the *Macworld* community. We'll also pay \$25 to \$100 for each *Open Window* item published.

Our first contribution to *Open Window* comes from

Macworld Contributing Editor Lon Poole. This hands-on application demonstrates one of the Mac's unique features—integrating text and graphics. Figures 1 through 6 illustrate the steps involved in setting up a letterhead using *MacPaint* and *MacWrite*. See the reviews of *MacPaint* and *MacWrite* in the Review section for a more comprehensive explanation of the features in these programs.

Send your Macintosh discoveries to *Open Window*, 555 De Haro St., San Francisco, CA 94107, or electronically to *CompuServe 74055,412 or The Source STE908*.



Figure 1

The logo image is created using *MacPaint*. The gallery name is in 18-point bold outlined New York type and filled with different patterns. The grid option is used to keep the text aligned.



Figure 2

The background pattern for the logo is created by editing one of the standard *MacPaint* patterns.

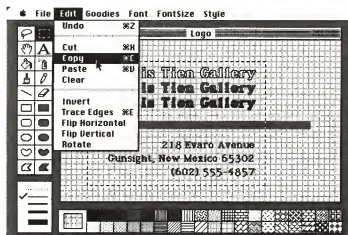


Figure 3

The selection rectangle is used to select the logo image. Choosing the Copy command stores a copy of the logo in the Clipboard. After saving the logo document, you can exit the MacPaint program.

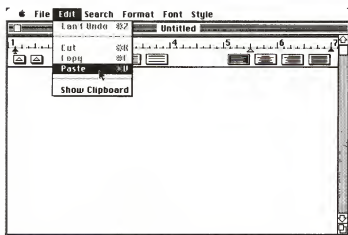


Figure 4

Opening a MacWrite document and choosing the Paste command from the Edit menu places the logo image in the document.

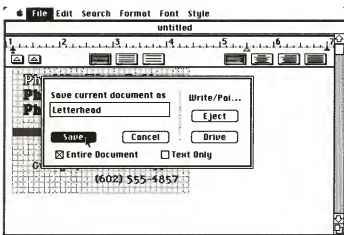


Figure 5

Clicking on the logo image allows you to reposition it in the document. The MacWrite document with the logo is then saved with the title Letterhead.

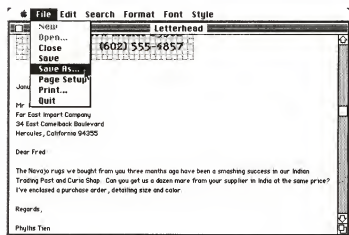
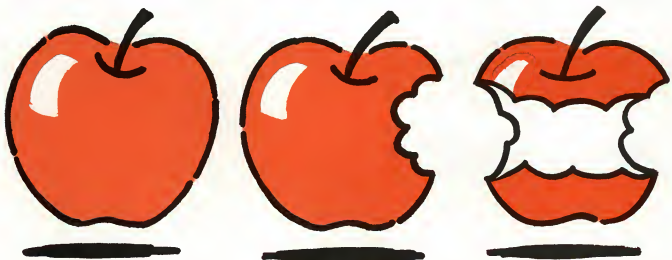


Figure 6

To write a letter using the letterhead, you would open the MacWrite document entitled Letterhead. If you wanted a permanent copy of a letter, you would choose Save As... and save the letter under a unique name.

Learn to use your Macintosh almost as fast as you can eat one.



Soon after Macintosh joins the Apple family tree we'll have an ATI training program ready to teach you how to use it.

ATI's "How to Use Your Macintosh" training disk and handbook will join our rapidly growing "how-to" series. We'll also have training programs for all the best software available for Macintosh.

All ATI training programs take you through the basics, step-by-step. Our interactive simulation method lets you practice as you learn. It's simple and fun!

Congratulations Macintosh! Together we'll help users- and dealers- bear fruit fast.

Contact ATI, 12638 Beatrice Street,
Los Angeles, CA 90066 (213) 546-5579.
Dealers: Call (213) 546-4725.



ATI Training Power™ is a trademark of American Training International



ATI Training Power™

Learn software the natural way.



**“GET SOPHISTICATED MAINFRAME
COMMUNICATIONS FOR MY MACINTOSH—
EVEN THE CAPABILITY TO SEND,
RECEIVE AND PROCESS DATA FILES?”**

“YOU CAN WITH PERSYST.”

Persyst communications products are already proven on the IBM PC®. Now they bring powerful new capability to your Macintosh®.

With Mac/3270™ and Mac/3770™, you can link your personal computer to mainframe computers. Plus receive, transmit and process data files.

For sophisticated communications and expansion products for the Macintosh, insist on Persyst.

PERSYST™

Persyst Products, Personal Systems Technology, Inc.
15801 Rockfield Blvd., Suite A, Irvine, CA 92714.
Telephone: (714) 859-8871. Telex: 467864

The Making of the Macintosh

Edited by Andrew Fluegelman

If you could have gone behind the scenes at Apple Computer while the Macintosh was being created, you would have been impressed by two things. One would have been the incredible complexity of the tasks that go into conceiving, building, testing, and marketing a major new computer. Some features that stand out in the final product may have been the result of a moment's inspiration, while others that are invisible to users were the result of months of hard work. Such is the process of building a computer; in that regard, the making of the Macintosh was not unique.

However, the other thing that would have impressed you was special: the deep personal commitment and conviction shared by everyone working on the project. The level of involvement approached religious fervor at times. There was endless deliberation and debate with never a hint of self-doubt. The motivation of these “hardware wizards” and “software evangelists” could not be explained by mere job descriptions. The people on the Macintosh project were driven by the unanimous belief that they had an opportunity to change the ways people work and think.



In late November 1983, as the final touches were being put on the Mac under round-the-clock schedules, I talked with some of the people who played key roles in the project. The following remarks are edited from tape-recorded conversations with Mac team members. Here are the personal reflections of those who made the machine.



Mike Boich

When I first came to the Macintosh project, I had nothing to do with software; I was involved in defining and marketing the Apple label products and building relationships with outside companies. Since then my role has evolved almost totally into "software evangelism," which means making sure that the smart, independent developers of the world, who have made the Apple II so successful, invest their best energies in the Macintosh. Showing the Macintosh to any sharp software developer is easy—there's really no salesmanship needed.

In the early days, before the documentation was complete, my job was mainly to support the independent developers and try to determine who the early developers would be. Prototypes were scarce and so was the time to support work on them. We decided, "Let's pick people who appreciate what the Mac is and encourage them to develop products that we think are appropriate for the machine."

We want to take functional applications and make them very accessible to people—accessible economically and in terms of how easy they are to use. We want to use the machine's power to do things out of the ordinary. Aesthetics are also an important factor in every application. The Macintosh is perhaps the first computer for which artists play a major role in designing the software.

Getting paid to share the Macintosh with people who appreciate it has been wonderful. Early on maybe 20 people were in the group, and we couldn't show it or talk about it to just anyone. If you have a beautiful thing, you want to tell everyone about it. I was the person who got to show it to the most people and see their reactions. That in itself was a lot of fun.



Donn Denman

I've been working primarily on MacBASIC, which isn't going to come out until after the machine is launched. Recently, I've been concentrating on odds and ends—helping out with the first release of the software. I worked on one of the desk accessories, the NotePad, and lately I've been working on the printer application that will facilitate printing files from the Finder.

Getting last-minute details finished has been pretty frantic at times. There's a lot of pressure in creating software. By its nature, it's never really finished. You keep thinking of little things you could make better. Software is never done until you *have* to get it done and you make the final disks and shrink-wrap them.

The best part of this project has been freedom—the freedom to go out on a limb and design something radically different.



Andy Hertzfeld

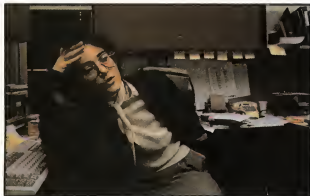
I started on the Macintosh project in February 1981 when there was just one tiny wire-wrapped Mac and all it could do was print "HELLO" about 85 times on the screen when you hit Reset. We were excited to see it do even that: "Wow! The computer actually does something!" For

the first six months of the project we worked on three things. First, we wrote routines to test the hardware to make sure that it worked the way it was supposed to. At the same time, we were getting a development system going, figuring out some way we could use the Lisa to write programs for the Mac.

The third thing going on at that time was writing demo programs to get people excited about the machine. This was before we even had QuickDraw. I actually wrote a mouse-based minisketch program that allowed you to draw with patterns and shapes. One of the more famous ones was the "Stretching Muppets." I took a screen portrait of the Muppets from the Apple II and wrote a little routine that was like a funhouse mirror that stretched the image very quickly. The image flowed like water down a waterfall, almost. I'd like to resurrect that someday.

We would do *hacks* that might seem frivolous, but they actually built excitement about the product. We had put together a team of sharp people, and we had to have something that would turn them on.

The early days of the project were the best for me, because it was us against the world. I'm an outlaw-type mentality. I like it when people say, "No, it's impossible to do that." That makes me want to do it. It's been amazing seeing this infant computer being born, watching it crackle with raw potential but with none of it really visible except to people who could imagine it, and then seeing it come to life over a period of years.



Joanna Hoffman

I started out as part of the research team because the Macintosh project originally was a research group. We were working on user interface design and on coming up with a new way of approaching users. As the project evolved, I started developing strategies for various markets. From the beginning I was interested in making sure that the product would be international.

One of the things we had to figure out was how to handle various data formats—dates, currency, and the like—because different countries handle these things in different ways. You can't simply do a straightforward translation—the conventions are basically different. We had to make sure that we could accommodate those various conventions properly.

We have a system that gives anyone who writes software for the machine the ability to write interna-

tional software. And it will all come out right at the end; it won't require redefining the code. A person will be able to sit down and edit the resource files on the screen, and then the data format and prompts will automatically appear in the right form and language for that country.



Jerry Manock

We have a five-person product design group that does industrial design, conceptualization, and layout and mechanical drawings. We handle all the tooling ourselves; we're a self-contained unit. As part of that group I essentially did all the drawings for the Macintosh, from layout to original

models to final production drawings.

Of course, we meant to do everything right the first time, but there were inevitable minor and major changes. The most significant change was to the 3½-inch disk drives halfway through the project. Getting the design updated in time was the big challenge because some major modifications had to be done on hard steel tooling. We told everybody that if we were to meet the schedule, the plastic would still be hot when it got to the assembly line, and that's essentially happening. But we've been lucky to make the schedule that we're committed to.

It's always exciting to see the first prototype working—not just with your parts in it, but with the software inserted and working. The first one that goes together and works is definitely a thrill. Then to see a hundred of them on the assembly line is neat—to see a thousand of them, ten times neater. But I think I can speak for all of us: the biggest thrill is to see the enthusiasm of people who discover the Macintosh for the first time. There are many subtle things about the design of the machine—little features that you probably would not notice until you'd had it for a week. And we all know about them. To see people discovering those features for themselves—to see them say, "Oh, now I know why they put this here. When I do this, it fits with that"—that's really exciting.



Dave Egner

I'm responsible for meeting the FCC requirements for radio frequency interference (RFI). Some people also call it electromagnetic interference. Those are just ten-dollar words that basically mean noise that comes in over your radio or TV set. The computer uses a 16-MHz crystal oscillator. To give you an idea, the AM radio band is from .5 MHz to 1.6 MHz. So 16 MHz is quite fast.

My job was to block out RFI. You can do it for altruistic or social reasons, or you can do it because the law says you have to, which is why most people do. The FCC has some rather rigorous requirements for home computers. It's interesting that the law is actually much stricter than you think. It's really on the side of the consumer in this respect.

It's never difficult to slap band-aids onto things. The difficulty is in meeting the requirements and keeping the computer inexpensive. If you look at some of the design band-aids that other companies have done, our solution turns out looking pretty good.



Barbara Koalkin

My job is product marketing manager, and my group is responsible for all the product-related activities that go on within the marketing base. We've worked closely with engineering to finish the product, test it, package it, document it, and figure out how to describe it in the product brochure and to the press.

We're working with third-party vendors to get some initial software and hardware peripherals, and working on launching the Mac in seven foreign countries as soon as possible after domestic introduction.

The major challenge has been trying to keep quality and excellence in everything we do for the Macintosh while also finishing it in a timely manner so that we can get the product out on schedule. Making the right trade-offs between timeliness and quality have been the most difficult decisions.

Many compromises have come up. "Should we bundle a piece of software or shouldn't we? Is it too late to change the manual? Does the Mac look exactly like what we want it to be, or should we add another stroke here?" We make decisions like these every day.

This is the best group of people I've ever worked with in terms of quality, excellence, and energy level. This may sound corny, but working on the Macintosh is really a once-in-a-lifetime opportunity.



Mike Murray

I'm the messenger of the project. My job is to understand what the engineers have done and to put that into words so plain folks like me will be able to understand it.

We faced some tough choices in making this product. We've had to focus on a few things and execute them very well.

We've had to make decisions between alternative A and alternative B. If we choose B and it's the wrong one, we're all going to feel bad.

In the process, our course has sometimes been 90 degrees different than the normal course people at Apple or other companies take in the way they do and think about things. We've had to be problem makers and then problem solvers. Often the problems we've made have been pretty big ones.

But what has gotten us through all this is the fact that we believe strongly in what we're doing and, I guess, in the romance of the product. We've had a love affair with this thing we call the Macintosh, and it's made us all work 90 hours a week.

For myself, the reward has been being able to stand up in front of groups of people, to talk to them about the Macintosh, and then to see them respond in a way that tells us that what we did was right. We aren't in the business of bringing out another type of aluminum siding or a new loaf of bread—we're doing something that's really going to change the way people do things.

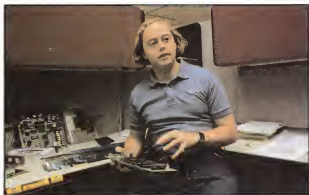


Susan Kare

My job is to make the software look nice. It's a high priority—having the aesthetics of this computer be as friendly, appealing, and precise as the capabilities allow. I'm responsible for the look of the windows and dialog boxes and also the fonts and icons.

Even though I didn't invent the machine or write the software, I feel a responsibility to have the Mac software look classy and be appealing without being cute and becoming tiresome. I look around at the icons and little symbols that exist on other systems and I think, "We have to have the best because a lot of people are going to be using our software for a long time."

I think the Mac will become a medium in its own right. Today most art done on computers is branded "computer art." That seems a little ludicrous—if it's really good art, you wouldn't have a computer art show, you'd have a print show. Some of those prints would be lithographs and some would be dot matrix. I think that the Mac can be used to produce quality graphics good enough to be sold commercially.



Burrell Smith

My official title is "Hardware Wizard." I started working on the Mac in the fall of 1979, in the very early stages. Jeff Raskin had initiated the idea of making an appliance computer. It had fixed functions and was tiny and versatile. At that point, Apple had been going in a different direction, and this appliance computer

was a very "researchy" thing to propose, so I got a system up and running quickly. It was an 8-bit machine that was incredibly tiny, but it didn't have enough CPU power.

Around January 1981 Steve Jobs got involved and said the Lisa group was doing amazing things with 68000 graphics. We came up with a clever way of hooking the 68000 CPU up to memory. This was our first effort. Although it was very tiny and had 384 or 256 dots of screen resolution and plenty of computing power, we wanted a few more things.

We wanted more memory. We wanted a little higher performance and better communications capability built into the machine. So we did another pass and produced what basically is the Macintosh that exists today.

Throughout the process, my goal was to take that idea of integrated functionality and make it into a real kicking, breathing, and screaming computer. Finally,

●●●●● *The Macintosh
is a breakthrough
because it allows both
computer experts and
computer novices to
share the technology.
—Burrell Smith*

we were able to pull it off. The biggest challenge was simply fitting it all in and having it do the things that we wanted it to do, such as having amazing sound capability.

For me, it also was a challenge, coming from my lowly background as a service technician. Being able to go from the very bottom of the company to a top technical position is the kind of thing that happens only at Apple.

The process I used to come up with these new designs was to prototype in my head. You can come up with an idea and you're able to analyze it to see whether that idea has any merit before you put your pencil on paper. It allows you to build thousands of prototypes—little mental creations that you can explore without having to build a breadboard and a thousand different things. If you had to sort through physical models, you'd be wiring forever.

People are going to think that the Mac is a breakthrough because it represents a critical point that allows both computer experts and computer novices to share the technology, each at their own levels of effectiveness. Previously, computers have been designed as machines that computer science people think ought to be good for other people.

With the Mac, we've been able to experiment enough to find exactly the right combinations—keeping the user interface consistent while maintaining the functionality of the machine. Many people will take a Mac out of the box, plug in the cables, put in a disk, and immediately be able to use the computer. That's something people have not been able to do before. With the Mac, we're saying, "It's okay to have power, but it's not okay to require people to know much more than they need to know to control the machine."

The key to the Mac is to spare people from having to know all the complexities of its innards. Computer people nowadays want to get things done—they want to concentrate on their problems. When they're presenting a graph, they want to be thinking about the little subtleties they want to portray. I think with the Mac you'll find that for the first time people truly will be concentrating on the problem itself and not on the computer hardware mechanisms.

Those of us who made the Mac have had to live in two worlds. As the designers and synthesizers of the machine, we have had to know every last detail about the system. But we also had to function as consumers—very fussy consumers. We all said, "What would I want to see the computer do? What would my kid brother want to see? What's the least amount of hassle that we can put people through and allow them to experience the joy of creating things that they never thought they could do?"



Chris Espinosa

I was mainly responsible for pulling together the Macintosh user interface guidelines documentation—the way of using windows so that all the applications resemble each other and work similarly. On many other computer systems, every application is written by a different person in a dif-

ferent place, and they each have their own ideas of how to make a computer work. That's terrific for the application writers, but not for users who have to learn every application and its way of using cursors, choosing commands, and selecting information.

With the Macintosh we wanted to determine a certain way to do all those functions, and then have everybody agree to do it that way. To hasten that agreement, we wrote the programs to do user interface things and put those programs in the ROM. They're

the User Interface Toolbox, and they make the Mac different from any other mouse-based computer. All the applications will work the same because we agreed on a way to build the Toolbox into every single machine.

Developing the user interface guidelines involved a lot of give-and-take and negotiation. It was about half design and half politics.



Jerome Coonen

I started in February 1983 as the section manager of the software group. In some ways my job is just to hold hands—to make sure everybody's happy and knows what to do. I like technical work, and once in a while I actually get to do something useful. I contribute something I have on paper or diskette somewhere, and in a sense that gives me the most pleasure. But that's not really my job—it's a luxury for me.

There were two big pushes on the software end. The first was finalizing the ROM, which we finished in September. The second was finishing the RAM-based operating system, which went out on diskettes in December. Each of those had a "drop-dead" date; my job was to make sure we hit all the milestones along the way. Sometimes that involved juggling people around, trying to get them to do the most critical task at any given time. Usually it was just a matter of coaxing—trying to assure my supervisors that everything was going along smoothly.

Scheduling software is almost impossible. I don't know anybody who knows how to do that. I certainly don't. When I started, I spent about two months trying to figure out a schedule for the ROM. It was a waste of time. We just had to keep looking every step of the way, trying to see if everybody was on track. Dealing with people is difficult when a lot of volatile, individual personalities are around, but that's the nature of any management job.

No other product looks anything like the Mac; people will be awed by it. But if we have bugs that make the programs crash, people will just notice that it crashes, no matter how good it looks.



Bob Belleville

I came to the project in April 1982. I had the job of trying to put in place all the pieces that would get the product on the market. I've been in the computer business more than 18 years. Many of my own designs never saw the light of day. They were perfectly good designs, but they didn't get out because there was no champion for the project, no way of working through the corporate maze and the corporate malaise. Even in a company like Apple, which is supposed to be lean and mean, small and well managed, it's hard to get a product finished.

There was a tendency among this group to keep refining the design—trying to make it perfect. Sometimes perfection is a little too far down the design scheme. I made it my goal to be sure that the fine work of all these people at Apple got out in a timely fashion.

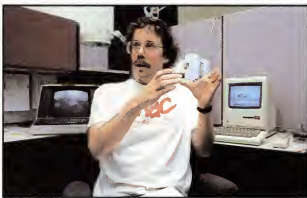


George Crow

Originally I was hired to design the power supply and the CRT portion of the machine. As the project matured I picked up responsibility for the disk drive. Initially, we had planned on using the same 5¼-inch disk drive as in the Lisa. To put it tactfully, we decided that the 3½-inch drive was the drive of the future, so in the middle of the project we switched over to the little 3½-inch hard shell disks. I've always been a strong proponent of them; in fact I was so strong for them that I wound up lining up original equipment manufacturers for the drives.

The lateness of the transition was the most critical part of installing the new drives. We even got as far as making hard tooling with a 5¼-inch slot. We had to change the bezel. We were really under the gun because we brought Sony in about May of 1983. Their drives have been super performers—they're really going to work out well for us.

Among the nicest things the customer will see is how easy it is to handle the disks.



Bill Atkinson

I was the main designer of the Lisa, and when we brought its technology over to the Macintosh, I came to the project to do an application. On Lisa I did only systems programs, but for the Macintosh I wanted to do an end-user application so that I could have more direct contact with the actual users. The result is *MacPaint*.

One of my central jobs has been to make sure that the Lisa and the Macintosh are compatible. Toward that end, the QuickDraw graphics package is 100 percent identical on both machines. Moving applications from one to the other is easy.

Designing *MacPaint* required a lot of iteration to make it simple—maybe 90 percent of the work was in designing the program as opposed to coding it. The program took a great deal of work to perfect—the first ten times I was simply willing to rewrite it.

I tried out versions of it on a number of people. When I saw they were confused and didn't understand it, I knew I must have the wrong model. I find that when you start getting an application really simple and clean, all the pieces start falling together.

Figuring out how to do the lasso was very important. Instead of describing the complex mechanisms underneath *MacPaint* that allow you to make a non-rectangular selection, I provided a simple metaphor to draw a ring around something: the way a lasso tightens around an object and grabs it. Underneath that simple action, *MacPaint* is going through a lot of processing.

If you keep the design team of a computer system very small, you can all be of one mind and spend less time arguing about what to build and more time working together to build it. When people were about to join the team, we showed them the Mac, and if they weren't excited enough to want to take it right home, they didn't belong on the team.

Watching people use the Macintosh and get excited, particularly people who aren't computer jocks, is rewarding. Computer jocks are thrilled because they realize how much work is behind it, but people without much knowledge of computers don't know that—they just love the way it works. They appreciate usefulness rather than technical achievement.

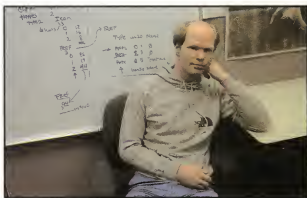


Steve Capps

I came on fairly late in the game, as a sort of designated janitor. I played clean-up everywhere.

When I first got here Andy Hertzfeld casually asked whether I could do the text editing package in 1.5K [1K = 1024 bytes]. He later admitted that he was willing to accept 2K but said 1.5 so that I'd make the 2K limit. When I first wrote the code, it was only 1K of Pascal, and I said, "Wow, I have it made, no sweat!" But it was too slow and therefore basically unusable. You might not think so, but creating a text editor is actually one of the most asymmetrical problems I've ever attacked. It turns out to be "special cases deluxe." It was very challenging and frustrating because I'd work hard and get nowhere. Then I'd put it away for about a week and try again.

Suddenly one day it was easy. I must have waked up or something, and I just got an idea of how to do it. It ended up being about 2400 bytes, but I'd never been in such a crunch mode. The result is the text edit package that's built into the ROM. It's the little routine that takes over when you're asked to type in a file name.



Bruce Horn

I've worked mainly on the software that you don't see very much: operating system things such as the file system, memory manager, drivers, basic I/O, and interrupts. I spent a lot of time on the software for the Sony disk, and before that the "Twiggy" [Lisa] disk. When we first got the Sony drives, we weren't sure we could read and write to them. The disk driver is a very technical program. You've got only about 16 microseconds to try to get a byte in. You've got to handshake everything. It's a matter of paying attention to detail. I worry a lot about interrupts. We get habituated to the machine because we're sitting here every day with it, but I think outside people are going to be blown away by its capabilities.



Larry Kenyon

I was hired originally to work on the Finder. In thinking about that and about some of the international considerations, I came up with the idea for the Resource Manager. It took awhile for everyone to accept it, but it's been pretty useful. The Resource Manager was done later in the project. As a result, various people had to lift the entire operating system up so that I could slide the Resource Manager underneath and then plunk down the rest and hope that everything worked.

The reason the Resource Manager is so useful is that it allows a program to be independent of any human language. It's the key to creating international versions of programs and the reason that we think the Mac will be an important machine in many countries.

The best part of this project for me has been working with really great people. The most difficult aspect has been giving up the other parts of my life to do it.

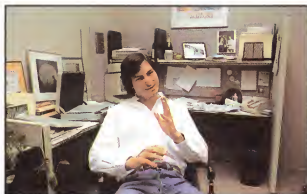


Rony Sebok

When I came here they had pretty much gotten things settled down, but there was a great deal of testing to be done because we were just about to freeze the ROM. I helped out with testing the ROM code.

Everyone's going to get a training disk with the Mac, and part of it teaches you how to use the mouse. So I wrote this cute program that makes you drag a mouse through a maze so that you learn mouse dexterity and how to point. There's a little magician and a bunch of hats. You point to a hat and out pop various nice graphics that Susan Kare drew.

I started on the project July 11, 1983, right out of Harvard. I had interviewed with about 12 companies, and someone suggested that I look at Apple. I talked to the Mac people, and I was turned on right away by this group. Everyone is really excited about the project. They're bright and dedicated to getting great computers out into the world. I think that will be reflected in the product.



Steve Jobs

The people who are doing the work are the moving force behind the Macintosh. My job is to create a space for them, to clear out the rest of the organization and keep it at bay. I can't spend enough time here, unfortunately, because I have other responsibilities. But every spare moment I have, I dash back because this is the most fun place in the world.

This is the neatest group of people I've ever worked with. They're all exceptionally bright, but more importantly they share a quality about the way they look at life, which is that the journey is the reward. They really want to see this product out in the world. It's more important than their personal lives right now.

The Apple II had a magical feel about it. You couldn't quantify it, but you could tell. The Macintosh is the second thing in my life that's ever felt that way. Opportunities like this don't come along very often. You know somehow that it's the start of something great. So everyone wants it to be perfect and works really hard on it. Everyone feels a personal responsibility for the project.

The Macintosh is the future of Apple Computer. And it's being done by a bunch of people who are incredibly talented but who in most organizations would be working three levels below the impact of the decisions they're making in this organization. It's one of those things that you know won't last forever. The group might stay together maybe for one more iteration of the product, and then they'll go their separate ways. For a very special moment, all of us have come together to make this new product. We feel this may be the best thing we'll ever do with our lives.

Macworld Gallery

*An exhibition of
Macintosh graphics*

Jeffrey S. Young, Curator

The Macintosh is certain to attract many people who are intrigued by the potential of Bill Atkinson's *MacPaint* graphics program. Professional artists and people who have never touched a paintbrush will be exploring the intricacies of the Mac's electronic easel. *Macworld Gallery*, a regular *Macworld* feature, will serve as a showcase for the talent and creativity of "Mac artists," exhibiting their unique artwork and sharing their comments about the techniques they used to create their drawings.

Macworld Gallery's first featured artist is Susan Kare, one of the inner circle of creative people who helped shape the Mac. Susan joined the Macintosh development team about a year ago as the "resident artist," and is responsible for creating many of the graphic images that make up the desktop environment as well as the type fonts used in *MacWrite* and *MacPaint*.

Susan has extensive training in the "traditional" fine arts, including a Ph.D. in Art History from New York University, but she had never used a computer before joining the Mac development team. She began working on an Apple II with a graphics tablet, quickly adapting her skills to the new medium and enjoying the flexibility the computer offers in creating drawings.

But the Mac and the *MacPaint* program are her real forte. The series of drawings that Susan has lent to *Macworld Gallery* demonstrate what is graphically possible with the Mac. As such, "Susan," as the Mac team calls her, is the ideal artist to introduce *Macworld Gallery*.

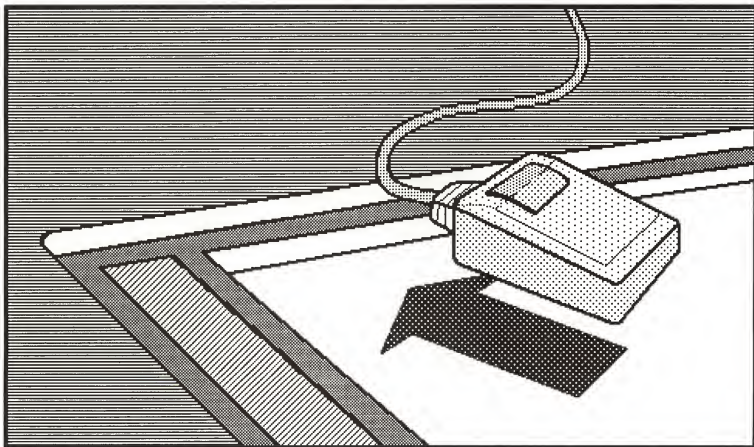


Oriental Girl

Oriental Girl

"Steve Jobs had a beautiful book of Chinese woodcuts, and we wondered what it would be like to make a facsimile. We ran this one through a digitizer (a kind of TV camera that produces a bit-map display of whatever it is focused on), and read the result onto a Mac disk. Then I sketched in the details to enhance the image.

"In the hair, for instance, where there are streaks of white in the black, I used the single-dot brush shape—which I've found to be one of the most useful tools—to paint white over the black. To create the flower design on the robe, I drew a single flower, peeled off several copies [holding



Mouse with Arrow

down the **⌘** and Option keys while you drag a selected section of a drawing “peels off,” or makes, multiple copies], and moved them around. Then I went back in and blurred the detail by painting in black with the single-dot brush.”

Mouse with Arrow

“I drew the Macintosh mouse to see if I could make a hard-edged, technical illustration; I wanted to get a transfer-film sort of effect. Every pattern in this drawing is from the default pattern palette, and every tool I used is from the standard tool palette. It’s just a simple, straightforward illustration that anyone could do with a little practice on the Mac.

“Another thing I’ve learned is that some things look great when xeroxed, while others don’t. For instance, black lines come out terrific, but gray is not so hot. After a little while you figure out what will work and what won’t.”

Knots & Vines

"You could have a disk with all sorts of ornamental details—architectural features, special patterns, logos, and so on. This drawing shows what you can do with a repeating design. I made up three sets of three leaves, then I peeled off a copy and flipped it, so instead of three originals, I had six. Then I filled the screen with copies and started moving them at random into rows.

"The knots in the border were done at the single-dot level with FatBits. I crossed the straight lines to form the outside, then lassoed the middle where they crossed, and tried flipping and inverting it. Then I worked on the knots in FatBits until I had the effect I wanted. Notice that I alternated the kind of knot at each of the crossings, because nobody ever ties a knot the same way twice."



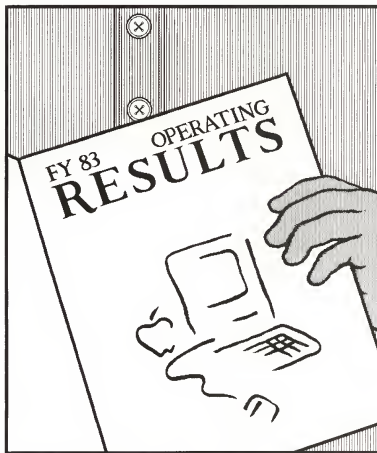
Knots & Vines

Macintosh Report Cover

"I wanted to make something that anyone could use to improve the look of reports, memos, or any type of office paperwork. It doesn't have to look sensational, just appealing. I was trying to make this look a little like a woodcut.

"The diagonal letters were sketched freehand with the pencil and then filled in with the paintbrush. I used FatBits to clean it all up. I change the pattern of the shirt every month, so that with just a slight variation I can make each report distinctive.

"The line of vertical stitching on either side of the buttons was done by clicking in a heavier vertical black line (using the Shift key to keep it straight), and then taking out one dot at a given interval. Look carefully in the center of the buttons, where the two strands of thread cross. I popped out a single dot on either side to give it the illusion of depth."



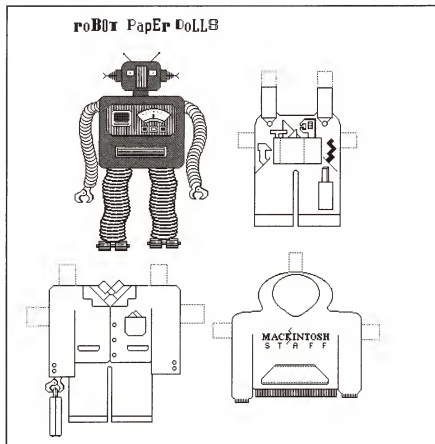
Macintosh Report Cover

Robot Paper Dolls

"I was just playing around, trying to come up with something that parents could do with their kids on the Mac. What if you had a disk with a hundred dolls, each with its own set of clothes patterns, and you could print up different sets for as long as you wanted to?"

"You'd simply paste a copy of the doll on the screen, and then start building the clothes around it. It would be easy for a child to use the paint bucket to fill in different patterns in the clothes. If the child had experience with the Mac, he or she could create personalized patterns with monograms."

"Putting something like this on disk could produce virtually inexhaustible toy supplies."

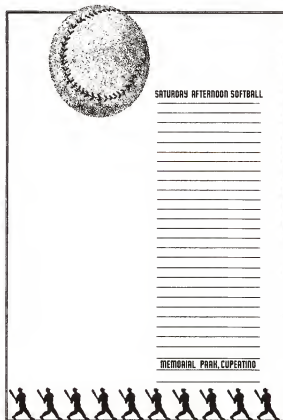


Robot Paper Dolls

Mac Sluggers

"This drawing was done when *MacPaint* was in its primary stages. There wasn't a spray can, so I had to create the baseball image dot by dot. I went to a meeting with the software people, and they thought the drawing was digitized. I took that comment as a compliment, because I was trying to make it look realistic."

"Many of the things that were painstaking to do at that time can now be done with ease. The stitching on the ball can be dragged around with the lasso. The 'sluggers' on the bottom row are created by copying the figure across the screen."



Mac Sluggers

Acme Detective Agency

"See that lettuce on top of the burger? I gave it that rippled, wavy effect by drawing in a pattern, lassoing one little section at a time, and moving it just a dot or two off. When the section drops back in place, it's skewed just a little, giving the drawing a more three-dimensional look. I also touched it up a bit with the spray can in white to give it an even more ripply effect.

"The rest of the drawing is variations on ovals. For the tomato I took an oval and then stretched it, filled it in, and sprayed just a bit of white from the spray can to give it some highlighting. The hamburger bun was made out of two half-ovals. I made two ovals, cut off the top of one and the bottom of another, and joined them together—that's why each has a different shape. Then I made a copy and, with a few changes in perspective, I had another bun."

.....
To be considered for exhibition in Macworld Gallery, send a paper copy of your drawing and a short paragraph describing the techniques you used to create the drawing to Macworld Gallery, 555 De Haro St., San Francisco, CA 94107. Macworld pays \$25 for each drawing exhibited. If your drawing is selected for exhibition, we ask that you send a copy of it on disk.

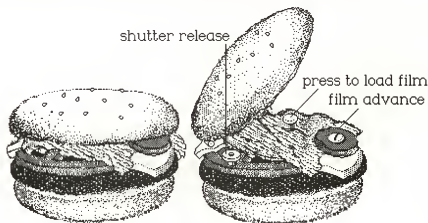
ACME Detective Agency

P.O. Box 2554 Altoona, PA

24 January 1984

Dear Sid:

Just got back from the Orient. Unfortunately, we never did find the missing jewels or the heiress. Spotted an interesting camera in the Bangkok airport gift shop, though--should be standard issue around ACME. Lightweight, waterproof, and a dead ringer for a cheeseburger.



You load the film (35mm) between the cheese and the lettuce--bite on the tomato to release the shutter. The pickle slice advances the film, the onion ring changes f-stops.

I think they'd be a heck of a lot more efficient than that saxophone camera Wilkerson picked up in Tuscaloosa.

Let me know what you think. Have to find my glasses and catch a plane.

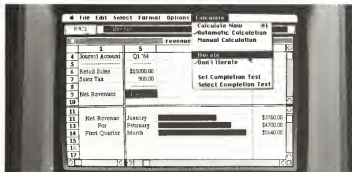
Regards,

319

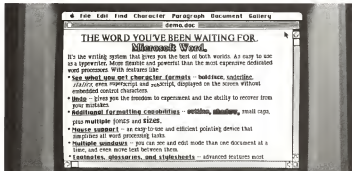
Agent 319

Acme Detective Agency

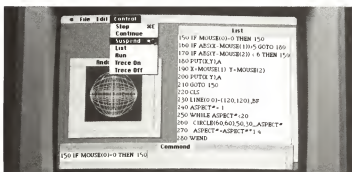
Microsoft's greatest hits are now playing on Macintosh.™



MULTIPLAN® The most powerful spreadsheet on the market.



WORD. Most dedicated word processors can't touch it.



MICROSOFT® BASIC. The industry standard. Plus special commands for the mouse and bit-mapped graphics.

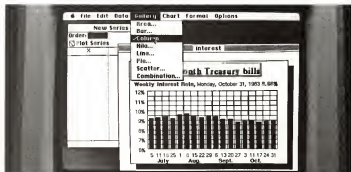
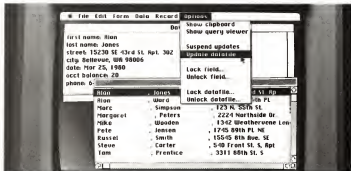


CHART. The first truly sophisticated and flexible graphics system for personal computers.



FILE. Finally, a data manager advanced enough to be simple.

Microsoft was founded on the idea that computers should be simple, easy and intuitive. The new Macintosh environment gives us a great opportunity to prove that idea.

We'll start with four of our high performance programs to make the most of your Mac. And there are more to come. Lots more.

Pioneering like this is nothing new for us. We wrote the first microcomputer BASIC in 1975 and developed the operating system for the IBM® PC.

If you'd like to make the most of your Mac, call 800-426-9400 (in Washington state, call 206-828-8088) for a free brochure and the name of your nearest Microsoft dealer.

**MICRO
SOFT**

Microsoft and Multiplan are registered trademarks of Microsoft Corporation.
IBM is a registered trademark of International Business Machines Corporation.
Macintosh is a trademark of Apple Computer, Inc.

Macworld Authors

David Bunnell, Publisher of *Macworld*, is the founder of *Personal Computing*, *PC*, and *PC World* magazines. He participated in the creation of the first personal computer, organized the first personal computer trade show, and published the first magazine with a bound-in floppy disk.

Matthew Douglas is a freelance writer specializing in microcomputer applications.

Andrew Fluegelman is the Editor-in-Chief of *Macworld* and *PC World*. He is the author of the popular communications program *PC-Talk* and coauthor of *Writing in the Computer Age*.

Lon Poole is a Contributing Editor of *Macworld* and the author of several computer books including *The Apple II Users Guide*.

Andrew Williams is a Contributing Editor of *Macworld* and *PC World* and the author of *WHAT IF... A Guide to Selecting and Using Electronic Spreadsheets on the IBM PC* and *The Know How Guide to SuperCalc 1, 2, and 3*, both forthcoming this year.

Jeffrey S. Young is a freelance journalist who has written for several national publications including *Esquire*.

Advertisers

American Training International, 12, 124
Apple Computer, 6-7
Davong, IBC
Dilithium Press, 103
Howard W. Sams, 48
Human Video, 4-5
Living Video Text, 33
Microtek, 31
Microsoft, IFC, 9, 141
Odessa, 49
Persyst, 47, 125
Software Publishing, 93
Tecmar, 1, BC
Verbatim, 8
Winterhalter, 121

Invitation to Writers

..... *Macworld* invites writers to submit articles for publication. Articles should be focused to fit within the editorial structure of the magazine.

▲▲▲▲▲ *Macworld* consists of four editorial sections: *State of the Art*, *Review*, *Hands On*, and *Community*. *State of the Art* covers developing technologies for the Macintosh. In *Review*, new Macintosh hardware and software are critically and objectively analyzed by experienced users. *Hands On* offers how-to articles that instruct readers on Macintosh applications.

Community articles describe the impact of the Macintosh in workplaces, schools, and homes.

————— *Macworld* accepts commentaries for the guest editorial department, *NotePad*. User group members are invited to submit articles on any facet of their organization.

..... Articles may be submitted by query or as unsolicited manuscripts to Editorial Proposals, *Macworld*, 555 De Haro St., San Francisco, CA 94107. If you send a query letter, please include an outline and a short description of the article you are proposing, a brief account of your computer-related experience, and a sample of your work.

Coming Attractions



Hands On

Communications Primer

Going on line with the world requires a number of special communications skills. We'll show you how to connect your Mac to the phone lines using a modem and explain the protocols computers use when they speak to each other. We'll also describe working routines to help you profit from each communications session and acquaint you with on-line services available to Mac owners.

The Business Graphics Gallery

This hands-on article will explain some of the tools available in programs such as *Microsoft Chart*. We'll present tips for preparing business graphics presentations, including advice on selecting chart styles and modifications and an explanation of analytical tools available through graphics and statistical manipulation.



Community

The Class of '84

Apple Computer has embarked on an ambitious program for "seeding" Macs in more than a dozen colleges and universities. We'll describe the introduction of the Mac on campus and the ways different college administrators are implementing the plan. We'll speculate about the impact that this program may have on the way students learn, study, and communicate.

The Electronic Auditor

Peat Marwick Mitchell and Co. developed an auditing program, *SEACAS*, that runs on the Macintosh. The company plans to send two thousand of its senior auditors out on consultations equipped with a Mac and *SEACAS*. We'll examine the goals and strategies for using the program to computerize accounting tasks previously performed with calculators and pencils.

Plus

The Mac's hidden function keys
The future of data base management
Digitizing images for the Mac
A review of Microsoft Word
Tips on working with a single disk drive system
An interview with Bill Atkinson, creator of MacPaint
Tips and suggestions from readers in Open Window
An exhibition of Macintosh art in Macworld Gallery
Macintosh product descriptions in Macware News

The following summaries describe a few of the articles to be featured in upcoming issues of Macworld.

State of the Art

The Lisa Connection

On the same day that the Macintosh was launched, Apple announced a new version of its Lisa computer—Lisa 2—that is designed to run all the Macintosh software. We'll take a look at the Mac's "big sister" and evaluate what you can do with 1 megabyte of memory and 10 megabytes of Winchester disk storage.

In Search of the Finder

Have you wondered what rearranges icons on the desktop, asks you to switch disks, and generally keeps track of what's going on while you work on the Mac? The answer is the Finder. We'll discover where it "lives," investigate how it works, and suggest ways to make the Finder work for you.

Review

MacTerminal

The Mac's link to the world is the serial port that lets you communicate with computers of every stripe, including other Macs. The software that implements this capability is a communications program. We'll review Apple's version of this essential software, *MacTerminal*.

Microsoft Chart

Business graphics are now accessible to all Mac users through *Microsoft Chart*, the first graphing and plotting program available on the Macintosh. We'll present an in-depth analysis of this program's powerful features that translate numeric data into informative graphics presentations.



Introducing MACWORLD

The Macintosh Magazine

**The Exciting New Magazine for
the World's Most Creative Personal
Computer—the *Macintosh***

**At Special Charter Subscriber Rates
Save Up to 54% Off the Newsstand Price***

(*when you subscribe for 3 years)

Personal computing took a giant leap into the future when Apple introduced the Macintosh. We've created *Macworld* as your passport into this incredible new realm of computing. Each issue we'll bring you everything you need to explore and get the most out of your Macintosh. From the latest product news to innovative business applications, from amazing graphics to personal productivity tools and games, each issue of *Macworld* will be packed with interesting, practical, readable information written by skilled, perceptive writers.

Each month we'll be creating a *Macworld* community; sharing ideas, problems, and creative solutions while we explore the world of Macintosh together.

So why take a chance on missing a single exciting issue? For a limited time only you can subscribe to *Macworld* at special Charter Subscriber rates of only

 **\$24 for 1 year (12 issues)**

That's a savings of 20% off the regular \$30 subscription rate and 50% off the newsstand price!

**And you can save even more by subscribing
for 2 or 3 years at Special Charter Rates. Don't
Wait...Subscribe Today and Save!**

Meet Your New Business Team. Macintosh & Davong.



Apple's new Macintosh™ is the most exciting development in personal computing since the original Apples started it all.

Teamed with Davong peripherals, the incredibly easy to use Macintosh becomes a powerful, flexible and expandable business tool.

Available at more than 1,000 Davong dealers nationwide is the new Davong Mac Disk system, providing from 5 to 32 megabytes of hard disk storage. Mac Disk gives you the increased capacity and enhanced functionality you need for your business applications. All at a very reasonable price and with no sacrifice in the speed and ease of use built into your Macintosh.

And coming very soon from Davong are a 28 megabyte streaming tape backup system, for fast and secure backup of your files, and a multi-tasking local area networking system to allow all of the Macintoshes in your organization to share information, programs, and the advantages of Mac Disk.

See your local Davong Dealer today for details.



Davong Systems, Inc.
217 Humboldt Court
Sunnyvale, CA 94089
(408) 734-4900
Telex: 176386



Apple and Macintosh are trademarks of Apple Computer Inc.

TECMAR SWEETENS APPLES



Tecmar supports the MacIntosh[®] with a full line of Candy Apple[™] peripherals. See page 1 for further information.

*Trademark of Apple Computer, Inc.

TECMAR

Tecmar Inc. 6225 Cochran Road Solon (Cleveland), Ohio 44139-3377 Phone: (216)349-0600 Telex:466692